SecureDrop Documentation

Release latest

SecureDrop Team and Contributors

Jul 02, 2026

INTRODUCTION

SecureDrop is an open-source whistleblower submission system that media organizations can use to securely accept documents from and communicate with anonymous sources.

This documentation is intended for sources, journalists, and administrators. If you would like to contribute to SecureDrop, please see our developer documentation.

Note

This documentation is also available as a Tor Onion Service at http://dftlffjdogaragaxkc6jqxpo77s7rrngimyoq7uuq3clowhmttblcoyd.onion/en/stable/.

CHAPTER

ONE

GET STARTED

I want to learn more about how SecureDrop works.

I have information I want to share, and would like to learn how to do so safely.

I am looking to set up a SecureDrop installation.

I have a SecureDrop installation and am interested in next steps.

I am a journalist and would like information about how to best use this system.

Note

The terms in italics are terms of art specific to SecureDrop. The *Glossary* provides more-precise definitions of these and other terms. SecureDrop is designed against a comprehensive *threat model*, and has a specific notion of the *roles* that are involved in its operation.

1.1 What Is SecureDrop?

SecureDrop is an open-source whistleblower submission system that media organizations can use to securely accept documents from and communicate with anonymous sources.

1.1.1 Purpose

In many of the recent leak prosecutions in the United States, sources have been investigated because authorities are able to retrieve both metadata and content of communications from third parties like email and phone providers in secret. SecureDrop attempts to completely eliminate third parties from the equation so that news organizations can challenge any legal orders before handing over any data.

SecureDrop also substantially limits the metadata trail that may exist from journalist-source communications in the first place. In addition, it attempts to provide a safer environment for those communications than regular corporate news networks, which may be compromised.

Another key feature of SecureDrop is that *Journalists* can receive submissions from unknown sources without risking the security of their own machines and networks.

1.1.2 How it works

Sources and *Journalists* connect to SecureDrop using the Tor network. The SecureDrop software is running on premises on dedicated infrastructure (two physical servers and a firewall).

The following steps describe how a SecureDrop submission is submitted, received and reviewed:

1. A *Source* uploads a submission to the news organization using Tor Browser.

2. A *Journalist* connects to SecureDrop using their *SecureDrop Workstation*, where *Journalists* can view the document, process it (e.g., to remove metadata or potential malware), print it, or export it to a dedicated device.

See also

Check out *What makes SecureDrop Unique* to read more about SecureDrop's approach to keeping *Sources* safe.

1.1.3 User roles

There are three main user roles that interact with a SecureDrop instance:

Sources

A *Source* submits documents and messages by using Tor Browser (or Tails) to access the *Source Interface*: a public *Onion Service*. Submissions are encrypted in place on the *Application Server* as they are uploaded.

Journalists

Journalists working in the newsroom use a *SecureDrop Workstation* to connect to their SecureDrop to communicate with *Sources*. *Journalists* download GPG-encrypted submissions. Apart from those deliberately published, decrypted documents are never accessed in an Internet-connected environment.

Admins

The SecureDrop servers are managed by a systems admin; for larger newsrooms, there may be a team of systems admins. The admin connects to the *Application* and *Monitor Servers* over authenticated Onion Services, and manages them using Ansible.

1.1.4 Project history

The web application, which was originally called DeadDrop, was developed by Aaron Swartz in 2012 before his tragic death. The hardening guide and security environment was architected by James Dolan. Investigative journalist Kevin Poulsen originally managed the project. The New Yorker launched the first implementation and branded their version StrongBox in May 2013.

In October 2013, Freedom of the Press Foundation took over management and development of the open source project and re-named it SecureDrop. In the project's early years at FPF, development was driven by James Dolan and Garrett Robinson. Today, SecureDrop is maintained by a small full-time development team at FPF and a growing volunteer community.

1.1.5 Technology and contributions

SecureDrop and SecureDrop Workstation are open source projects of Freedom of the Press Foundation (FPF), a US-based nonprofit organization. You can support our work by contributing to SecureDrop and by making a donation.

Our work would not be possible without the larger open source community.

Tor provides the foundation for the the anonymizing network that allows *Sources*, *Journalists*, and administrators to maintain their privacy while connecting to SecureDrop.

We're deeply grateful to the SecureDrop volunteer community for translating our software into many languages. Their work is enabled by Weblate, an open source platform for continuous localization. You can make a donation to support Weblate development.

Translation of SecureDrop is supported by Localization Lab. You can donate to support their important work to help bring open source software into many languages.

The backbone of SecureDrop Workstation is Qubes OS. FPF has directly sponsored Qubes OS development, and we encourage you to donate to Qubes OS as well.

We use the Python programming language and many tools in its ecosystem, which you can support by donating to the Python Software Foundation.

SecureDrop Workstation VMs are powered by Debian and Fedora both of which rely on volunteer contributions and financial support. The GNOME project acts as an umbrella for many of the individual software components we rely on.

Finally, SecureDrop Workstation relies on many other open source projects such as Ubuntu Server, grsecurity, GnuPG, Sequoia, LibreOffice, Audacious, OpenPrinting, Apache, OSSEC, and others. These projects, in turn, are built on open source foundations. Please consider directing time and financial support wherever it can make a positive difference.

1.1.6 Privacy

The SecureDrop web interface does not record your IP address, information about your browser, computer, or operating system. Furthermore, the SecureDrop pages do not embed third-party content or deliver persistent cookies to your browser. The server will only store the date and time of the newest message sent from each source. Once you send a new message, the time and date of your previous message is automatically deleted.

Journalists are also encouraged to regularly delete all information from the SecureDrop server and store anything they would like saved in offline storage to minimize risk. More detailed information can be found in our *sample privacy policy*, which we encourage news organizations using SecureDrop to adopt from when creating their own. Make sure to also follow our *best practices for creating the SecureDrop *Landing Page** so that it logs as little information as possible as well.

1.1.7 Security

While we can't guarantee 100% security (no organization or product can), the goal of SecureDrop is to create a significantly more secure environment for *Sources* to share information than exists through normal digital channels. Of course, there are always risks. That said, each release of SecureDrop with major architectural changes goes through a security audit by a reputable third party security firm.

1.1.8 Audits

Before major code changes are shipped, our policy is to have SecureDrop audited by a professional, third-party security firm. You can find a list of all audits completed so far.

In addition to these audits, we also have a bug bounty program hosted by Bugcrowd.

1.1.9 Cost

SecureDrop is a free and open source application that costs nothing to install. However, the application does require hardware that news organizations must purchase, including two servers, several USB sticks, an air-gapped computer, and a firewall.

We have created a *recommended hardware guide;* following these recommendations wherever possible will minimize incompatibility risks. We are aiming to offer a set of recommendations that work for organizations at different scales.

It is critical that the hardware is owned by the media organization and stored on its premises in a secure space.

The total cost of the hardware we recommend is $2,200 to $2,400, though it can be done for less if you are willing to sacrifice size and speed on the servers or are able to use recycled machines sourced from within your organization.

As part of priority support agreements and on a pro-bono basis for smaller news organizations, Freedom of the Press Foundation will visit your offices, help set up SecureDrop and train *Journalists* to use it. (For pro-bono support, we request that our travel costs are covered.)

1.1.10 Environment overview

Server infrastructure

At SecureDrop's heart is a pair of servers: the *Application ("App") Server*, which runs the core SecureDrop software, and the *Monitor ("Mon") Server*, which keeps track of the *Application Server* and sends out alerts if there's a problem. These two servers run on dedicated hardware connected to a dedicated firewall appliance. They are typically located physically inside the newsroom, and must be physically located on-site within your organization's premises.

- ***Application Server*:**
  An Ubuntu server running two segmented Tor hidden services. The *Source* connects to the *Source Interface*, a public-facing Tor *Onion Service*, to send messages and documents to the *Journalist*. The *Journalist* connects to the *Journalist Interface*, an authenticated Tor *Onion Service*, to download encrypted documents and respond to *Sources*.
- ***Monitor Server*:**
  An Ubuntu server that monitors the *Application Server* with OSSEC and sends email alerts.

The servers connect to the network via a dedicated hardware firewall.

SecureDrop Workstations

The SecureDrop environment consists of at least one laptop, in addition to the servers described above:

- ***SecureDrop Workstation:***
  The laptop used by *Journalists* to download encrypted documents and respond to *Sources*, and used by administrators to perform maintenance on the servers.

1.1.11 Operation

Planning & preparation

Setting up SecureDrop is a multi-step process. Before getting started, you should make sure that you're prepared to operate and maintain it. You'll need a systems admin who's familiar with Linux, the GNU utilities, and the Bash shell. You'll need the *hardware* on which SecureDrop runs — this will normally cost $2000-$3000. The *Journalists* in your organization will need to be trained in the operation of SecureDrop, and you'll need to publish and promote your new SecureDrop instance afterwards — using your existing websites, mailing lists, and social media.

It is recommended that you have all of this planned out before you get started. If you need help, contact the Freedom of the Press Foundation who will be glad to help walk you through the process and make sure that you're ready to proceed.

Technical setup

Once you are familiar with the architecture and have all the hardware, *setting up SecureDrop* will take at least a day's work for your admin. We recommend that you set aside at least a week to *complete and test* your setup.

Provisioning & training

Once SecureDrop is installed, *Journalists* will need to be provided with accounts, two-factor credentials, workstations, and so on — and then trained to use these tools safely and reliably. You will probably also need to train additional backup admins so that you can be sure that your SecureDrop setup keeps running even when your main admin is on holiday.

Introducing staff to SecureDrop takes half a day. Training a group to use SecureDrop proficiently takes at least a day — and a single trainer can only work with so many people at once. You will probably need to run several training sessions to instruct an entire newsroom. Depending on staff availability, training and provisioning may take a week or more. If you have multiple offices, training will need to happen at each location. Again, the Freedom of the Press Foundation are happy to help you plan and train your team.

Going public

Once you have a SecureDrop instance and your team knows how to use it, you should test it thoroughly and then tell the world. The Freedom of the Press Foundation are happy to help you check that your SecureDrop setup is up-to-code and properly grounded. After that you'll want to check out the *best practices* for your SecureDrop *Landing Page* and our guide to *promoting your SecureDrop instance*.

1.1.12 Sharing access

With other *Journalists* in your organization

While SecureDrop supports having multiple journalist accounts for the document interface, all accounts will access the same inbox. To avoid confusion, we recommend news organizations assign 1-3 *Journalists* to regularly check SecureDrop and make sure that they all are in contact as to who is responsible for responding to each *Source*.

With other organizations

Currently you cannot use SecureDrop with multiple organizations for security reasons. One of the benefits of SecureDrop is that it completely eliminates third parties from your communication channel. The media organization owns and operates the server that both the *Source* and *Journalist* connect to.

Any legal request or order has to be served on the media organization operating the SecureDrop server, giving them a chance to challenge it before handing over any data. If a third party operated a SecureDrop server which multiple organizations used, a legal order could be served on the operator without the media organizations knowing.

1.2 What makes SecureDrop unique?

SecureDrop attempts to solve or mitigate several problems journalists and sources have faced in recent legal investigations, attacks from state actors, and other threats to the confidentiality of communications.

1.2.1 No third parties that can secretly be subpoenaed

For decades, there were very few leak prosecutions in the United States in large part because the government would have to subpoena reporters to testify against a source to get a conviction. That proved incredibly difficult, if not impossible, when reporters regularly refused to testify and threatened to go to jail rather than betray a source.

More recently, there have been a record number of leak prosecutions largely because the government has learned they don't need reporters to testify against their sources anymore. Instead, they can just secretly subpoena third-party services like Google or AT&T or Verizon or Facebook and get a treasure trove of digital information on reporters and sources' communications. For example, the Associated Press had twenty of their phone lines subpoenaed without their knowledge in order to identify a source. The government also got a warrant for Fox News reporter James Rosen's Gmail account without him knowing. In both cases, their alleged sources were prosecuted, even though journalists never directly divulged their sources.

SecureDrop completely eliminates third parties from the equation and puts the power to challenge such cases back in the hands of reporters. The journalist and source communicate exclusively through one server that the news organization owns and sits on their property, so any legal order for information must go directly to the news organization rather than Google or AT&T. The news organization again has the power to contest the order or refuse to comply if they so wish.

1.2.2 Limits the metadata trail as much as possible

In many leak cases, the metadata of a journalist's communications—where you're located, who you're talking to, when you're talking to them, and how often—can lead to trouble just as much as the actual content of your conversations.

Even if a government serves a court order directly to a news organization to compel the disclosure of information, SecureDrop logs much less information than email providers or phone companies do.

The source can only log into SecureDrop through Tor Browser, which masks the source's IP address to begin with, so there is no indication who the source is (unless they disclose it) and where they are sending information from. The Tor IP address, the computer, and the browser type that the source is using is not logged either.

For each source, only the time and date of each submission is logged on the server. When a source sends a new message, the time and date of the last message is overwritten. This means that there won't be a trail of metadata showing exactly when the source and journalist were talking.

In addition, sources cannot create a custom username that could reveal information about them. Instead, SecureDrop automatically generates two random codenames, one to show to the source and another to the journalists using the system.

1.2.3 Encrypted and air-gapped

Communications through SecureDrop are both encrypted in transit, so messages cannot be easily intercepted and read while they are traversing the Internet and are also encrypted on the server so if any attacker manages to break into the server, they would not be able to read past messages.

In addition, the decryption key for SecureDrop submissions sits in an isolated virtual machine inside a hardened operating system that opens submissions in a temporary, non-networked environment.

1.2.4 Protects against hackers

A 2014 study showed that 21 of the top 25 news organization had, at one time or another, been targeted by state sponsored hackers.

Because of this threat, SecureDrop completely segments its traffic from a news organization's normal network. Submissions are accessed and downloaded using the Qubes operating system, with all SecureDrop-related traffic routed through Tor.

The SecureDrop servers also undergo significant system hardening in order to make it as difficult as possible for hackers to break in. By doing so, SecureDrop protects sources against networks that are already compromised, as well as a news organization's normal network from attacks that could potentially come through SecureDrop.

1.2.5 Free and open source software

100% of SecureDrop's code is free and open source. Not only does this mean anyone can install SecureDrop themselves, but the code is available online for security experts to test for vulnerabilities.

SecureDrop has gone through multiple audits by third-party penetration testing firms and will continue to go through audits when major changes are made to the code base in the future. We always publish these audits publicly so everyone can be assured that SecureDrop is as safe to use as possible.

1.3 SecureDrop Workstation and Qubes OS

1.3.1 What is SecureDrop Workstation?

A SecureDrop Workstation is a laptop used by a *Journalist* to connect to a SecureDrop instance and securely view submissions and reply to messages from *Sources*. The SecureDrop Workstation is based on Qubes OS and it consists of several different carefully-configured virtual machines (VMs), so that everything a *Journalist* needs to use SecureDrop resides on one computer.

Encryption and decryption happen with one click using a network-isolated VM that holds the SecureDrop *Submission Private Key*. Submissions can be viewed securely on the same machine thanks to a feature of Qubes that creates temporary VMs in which to view untrusted content without exposing the rest of the system to that content. *Journalists* use the SecureDrop Workstation to decrypt, view, reply to, and export submissions.

A key feature of SecureDrop is that *Journalists* can receive submissions from unknown *Sources* without risking the security of their own machines and networks. Previously, SecureDrop accomplished this by using a physical airgap (the *Secure Viewing Station*); to view submissions, *Journalists* would have to download them, transfer them to an encrypted USB flash drive, and physically take that drive to a separate, non-networked computer for decryption and viewing. SecureDrop Workstation combines all of those steps into one workflow on one machine: a Qubes computer that combines the *Journalist Workstation* and the *Secure Viewing Station*.

1.3.2 What is Qubes OS?

Qubes OS is an open source, security-focused operating system. It is very different than operating systems you may be familiar with already, because it consists of multiple isolated virtual machines that allow you to separate more trusted components, files, or programs on your computer from less trusted components, files, or programs.

Broadly speaking, this means that even if files in one of your virtual machines are exposed to malware, files in others still have some protection, which is not true of other operating systems.

For more about the security features of Qubes, see the Qubes OS documentation.

1.3.3 SecureDrop Workstation networking architecture

One key security feature of Qubes OS is that it enables users to configure the appropriate level of network access for each VM. For example, you could have a VM for password storage that has no network access, a work VM that is firewalled to only connect to work servers, and a personal VM that always uses Tor.

SecureDrop Workstation tightly controls access to the network, in order to prevent the exfiltration of messages, replies, documents, or encryption keys by adversaries. Specifically, the following VMs have no network access:

- `sd-app`, which runs SecureDrop Inbox, and holds decrypted messages, replies, and attachments.
- `sd-viewer`, which is the template for disposable VMs used for opening and viewing attachments.
- `sd-gpg`, which holds the *Submission Private Key* required to decrypt messages, replies, and documents.
- `sd-devices`, which passes exported documents through to USB devices like printers and encrypted USB flash drives.

By design, the Qubes OS host domain, `dom0`, also does not have Internet access.

Note

If you attempt to directly access the network in any of these VMs, it will not work. That is the expected behavior.

Because SecureDrop Inbox must connect to the SecureDrop *Application Server* in order to send or retrieve messages, documents, and replies, it can communicate through Qubes-internal Remote Procedure Calls (RPCs) with another VM, `sd-proxy`, which can only access the open Internet through the Tor network.

Like all networked VMs, `sd-proxy` uses the `sys-firewall` service to connect to the network, which is provided via `sys-net`. All three VMs must be running for SecureDrop Inbox to successfully connect to the server.

Important

The `sd-proxy` VM contains a sensitive authentication token required to access the SecureDrop API via Tor, and should not be attached to VMs that are unrelated to SecureDrop.

Installing additional software on the SecureDrop Workstation

Right now, the project is designed to make the *Journalist* experience easier by combining the functionality of the *Journalist Workstation* and Secure Viewing Station. The main focus is making sure that checking SecureDrop is easier and faster.

While we hope to add advanced tooling and document-processing options down the line, at this time we request that you do not change the configuration of the workstation or install additional software on it. If you have specific needs that you would like to discuss with us, please contact us via Signal, or send us a PGP-encrypted email at support@freedom.press.

Why can't I save or print from the Viewer VM apps?

When you view a file on SecureDrop Workstation, it is opened in a disposable VM that cannot access the network or any peripherals. The VM and all its data will be destroyed the moment you close the viewer application.

You can save files from a viewer application, but copies saved inside a disposable VM will be deleted when you close the application, and the changes will not be applied to the main copy of the file stored on your computer.

You cannot print from the viewer application, because it does not have access to peripherals. This prevents malware from exfiltrating data (e.g., via attached USB devices), and from targeting hardware-level security vulnerabilities.

You *can* print files directly from SecureDrop Inbox by clicking **Print** for a downloaded file, which will pass the file through to your USB printer without opening it in an interactive viewer application.

Why can't I copy and paste?

You should be able to copy and paste *within* any VM on the system, e.g., from one application running in `sd-app` to another.

Copy and paste between and to SecureDrop Workstation VMs is disabled for security reasons. The goal of this restriction is to minimize the risk of accidental pastes of sensitive content, and to reduce the attack surface for attempts to exfiltrate information.

Administrators can configure limited exceptions to this policy; please see the section *Managing Clipboard Access* of the admin guide for more information.

How is using Qubes different from using virtual machines?

Virtual machines that run on your Mac, Windows, or Linux machine (such as those created using VirtualBox, Parallels, and so on) are a "guest" on your machine, but still require a "host" operating system on top of which to run. These virtual machines are not designed as security tools; if the host OS is compromised, there are no protections for the guest OS, and some features (such as networking) allow communications between guest and host that can compromise the security of both.

In contrast, Qubes virtualization occurs at a lower level, under the Xen hypervisor. This means that virtual machines (VMs) in a Qubes environment can run operating systems that are independent of each other and are not reliant on a host OS.

In addition, these virtual machines can be used to quarantine specific functions of your computer. For example, network access is provided via two or more VMs, and you can control which applications or files have access to a networked environment by connecting to or disconnecting from these VMs.

Finally, Qubes is designed to make it more difficult for malware to remain on your machine. Each VM has read-only access to the root filesystem that provides its operating system, meaning that if a VM is infected with malware, it will be more difficult for that malware to persist across a reboot of that VM.

For more about the security features of Qubes, see the Qubes OS documentation.

How does the security of this system compare to using an air-gapped *Secure Viewing Station*?

The air-gapped *Secure Viewing Station* that is part of a SecureDrop setup offers strong protections against exfiltration of submissions or encryption keys by adversaries. It lacks important protections that SecureDrop Workstation provides. On the other hand, vulnerabilities in Qubes OS or Xen Hypervisor may have a greater security impact than vulnerabilities in Tails, the operating system used on a *Secure Viewing Station*.

A typical *Secure Viewing Station* USB flash drive may contain documents from multiple *Sources* and always contains the highly sensitive private key needed to decrypt them. An adversary who does manage to achieve a security compromise (e.g., through a vulnerability in a file viewer application) can access these other files, and may be able to exfiltrate them.

In spite of the air-gap, this may be possible through physical channels used to transfer files off the *Secure Viewing Station* (e.g., USB flash drives), or by motivating the *Journalist* to perform an unsafe action (e.g., scanning a QR code).

Because the air-gapped *Secure Viewing Station* has no Internet access, updates can only be performed using another computer and a USB flash drive. In practice, newsrooms may not update their *Secure Viewing Station* in a timely manner, which can significantly worsen its security posture.

In SecureDrop Workstation, any document received via SecureDrop is opened in a disposable VM that has no Internet access and no access to other files submitted via SecureDrop. The encryption keys are stored in a separate, networkless VM from the SecureDrop Inbox application.

Because SecureDrop Workstation has Internet access, updates can be applied automatically as soon as they are available. SecureDrop Workstation enforces this by downloading and applying updates before the user logs into SecureDrop.

SecureDrop Workstation uses hardware-assisted virtualization, which allows us to use custom kernels for its VMs. These custom kernels use the grsecurity patches which are also used on the SecureDrop servers, and provide additional mitigation against security vulnerabilities.

An attacker able to exploit vulnerabilities in Qubes OS or Xen-based bare metal virtualization (likely in combination with other vulnerabilities, e.g., in a viewer application) may be able to exfiltrate information directly to the Internet. Qubes closely tracks any security vulnerabilities that may impact it, and the automatic update mechanism helps to ensure that, in the event of a vulnerability, every SecureDrop Workstation can be patched as quickly as possible.

For further technical detail on design rationale and mitigations, please consult our design document.

1.4 Getting support

Whether you are interested in learning more about SecureDrop, looking for help with an installation, or needing assistance with an existing SecureDrop instance, there are several support options available to you.

Freedom of the Press Foundation offers direct *support via Signal*.

If you are unable to use Signal, you can always contact us by email at securedrop@freedom.press (PGP encrypted).

Additionally, there is also some level of *Community Support*.

Note

If your installation is up and running, we recommend that you submit your SecureDrop to the SecureDrop directory. This also serves as a first introduction to the SecureDrop team.

While we will provide technical assistance within reason and at our discretion, we encourage you to consider a paid support agreement to receive priority support, staff training, or installation help. Visit the Priority Support and Training pages on the SecureDrop website for more information.

1.4.1 Support via Signal

Because of the sensitive nature of SecureDrop-related communications, we prefer providing support via the encrypted platform Signal.

Once we have connected with you on Signal, you will receive notifications regarding SecureDrop releases and security advisories, and you will be able to contact us with detailed requests for technical support.

Initial onboarding

Please start by submitting a request through the SecureDrop Contact Form.

Please provide an email address so we can reply back to you. We'll review your request and decide how to respond. If we decide to offer you support, we will send you instructions for onboarding you into a Signal group for your organization.

Using Signal

Signal must first be installed on an Android or iOS device and a phone number is required to create an account. A multi-platform desktop application is available which can sync messages and contacts with the mobile application. We recommend using the official Signal website for links and detailed instructions to install Signal on your devices.

Freedom of the Press Foundation has several guides to using Signal:

- Signal for beginners
- Locking down Signal
- Understanding every one of Signal's identifiers

1.4.2 Community support

You can connect directly with the SecureDrop development team and the larger SecureDrop community using the SecureDrop Gitter channel.

Warning

Remember that the Gitter channel is public. **Do not post any sensitive information through public channels.**

1.5 SecureDrop for *Sources*

Warning

This source guide below is meant to illustrate the source submission process for journalists and newsrooms, and is not intended to instruct sources how to safely use SecureDrop. If you are potentially interested in sharing information to a news organization via SecureDrop, you should:

1. Start from a place with public Wi-Fi, like a coffee shop. Use a computer you own and control. Do not use a mobile phone or tablet. Never use a workplace computer or network.
2. Download, install, and open Tor Browser, then visit `howto.securedrop.tor.onion`

1.5.1 What is SecureDrop?

SecureDrop is a tool that news organizations and NGOs use that enables secure and anonymous communication between whistleblowers and *Journalists*. No personal information is collected; information submitted to SecureDrop is encrypted, and SecureDrop is not a "cloud" service. If you don't have sensitive information to send to a news organization, it may be okay to use a traditional methods such as phone or email when reaching out.

SecureDrop can accept both messages and individual file uploads (up to 500MB). If you have multiple files to submit, you may do that. As a *Source*, you can also return to receive follow-up correspondence with an organization, or to send additional information. Dozens of news organizations — from *ProPublica* to *The New York Times* — use SecureDrop to accept tips securely and anonymously.

To truly protect your anonymity, it is important for you to take some extra precautions in advance. This resource will describe things you can do to help protect your anonymity when using SecureDrop. Note that your Internet Service Provider, or ISP (e.g., Comcast/Xfinity, Cox, Wave, etc), may already have a record of your visit to this website, docs.securedrop.org.

1.6 Before you submit

1.6.1 What NOT to do

- DO NOT access SecureDrop on your employer's network.
- DO NOT access SecureDrop using your employer's hardware.
- DO NOT access SecureDrop on your home internet network.

1.6.2 What to do

- **DO** carefully read the remaining instructions, that will carefully step-through the reasons why we advise the above, and provide guidance to minimize risk when using SecureDrop.

1.6.3 Suggested devices for using SecureDrop

When sensitive disclosures such as government improprieties are involved, we suggest you buy a new computer and at least one new USB flash drive. You should only use cash to make those purchases.

Many time-saving features of computers and phones can easily compromise your anonymity: bookmarks, recommendations, synchronization features, shortcuts to frequently opened files, etc. Those reasons and more are why using a dedicated computer for whistleblowing activities can be safer.

To build an even stronger buffer for yourself, we recommend booting the computer into the Tails operating system (typically from a USB flash drive). Tails is specifically designed to run on your computer without leaving traces of your activity. This may take some additional technical steps, but it is safer and fairly simple to get started. Even if you choose to use a dedicated computer for SecureDrop that will never be used for anything else, Tails will help to avoid leaving traces of your activity on the computer's hard disk, in your ISP's logs, or on cloud services.

1.6.4 Choose the right location

Find a busy cafe you don't regularly go to and sit at a place with your back to a wall to avoid cameras capturing information on your screen or keystrokes. Be sure to also make any purchases while there (WiFi, tea, snacks) or on your way to the cafe (bus, train, gas) with cash.

1.6.5 Use Tor Browser

Each SecureDrop may **only** be reached through the Tor Browser. SecureDrop pages are only available as *Onion Services*—encrypted web pages that end in ".onion," and only the Tor Browser is able to open these pages.

Tor is an anonymizing network that makes it difficult for anybody observing the network to associate a user's identity (e.g., the computer's IP address) with their activity. Tor Browser can be downloaded from the Tor Project's website. Tor Browser is a modified version of the Firefox web browser that also includes features protect your security and anonymity. If there is a chance that visiting the Tor Project's website to download Tor Browser might raise suspicion, you have a couple of alternatives:

- If your mail provider is less likely to be monitored, you can send a mail to gettor@torproject.org with the text "linux", "windows" or "osx" in the body (for your preferred operating system) and a bot will answer with instructions.
- You can request to receive the Tor Browser bundle via the @GetTor_bot on Telegram.

While using Tor Browser on your personal computer helps hide your activity on the network, it will leave traces of its own installation on your local machine. Most operating systems keep logs, for example, any time an application is used. The sensitivity of the information you share and the capabilities of those who may not want you to share that information, should be considered when making these decisions.

Important

Tor protects your anonymity, but third parties who can monitor your network traffic can detect *that you are using Tor*. They may even be able to do so long after your browser session, using network activity logs. This is why we recommend using Tor Browser from a cafe you do not visit regularly.

1.6.6 Choose who to submit to

We recommend conducting all research related to your submission in Tor Browser. If you are unsure whether you are using Tor, you can visit the address https://check.torproject.org.

All organizations operating SecureDrop have a *Landing Page* that provides their own organization-specific recommendations for using SecureDrop. We encourage you to consider an organization's *Landing Page* before submitting to them.

Note

Each SecureDrop instance is operated and administered independently by the organization you are submitting to. Only the *Journalists* associated with that organization can see your submissions.

Most organizations make their SecureDrop prominently accessible from their main website's homepage (for news organizations, typically under sections called "Tips" or "Contact us"). You can also find an incomplete list of organizations accepting submissions through SecureDrop in the SecureDrop Directory maintained by Freedom of the Press Foundation.

Using Tor Browser, find the ".onion" address for the SecureDrop for the organization that you wish to submit to.

Tip

If the organization does have an entry in the SecureDrop Directory, we recommend comparing the address of the entry with the one on the organization's own SecureDrop *Landing Page*.

If the two addresses don't match, please do not submit to this organization yet. Instead, please contact us through the SecureDrop website, using Tor Browser. For additional security, you can use our onion address in Tor:

`sdolvtfhatvsysc6l34d65ymdwxcujausv7k5jk4cy5ttzhjoi6fzvyd.onion/report-an-error`

We will update the directory entry if the information in it is incorrect.

Once you have located the onion address, copy it into the address bar in Tor Browser to visit the organization's SecureDrop.

1.7 How to submit

Note

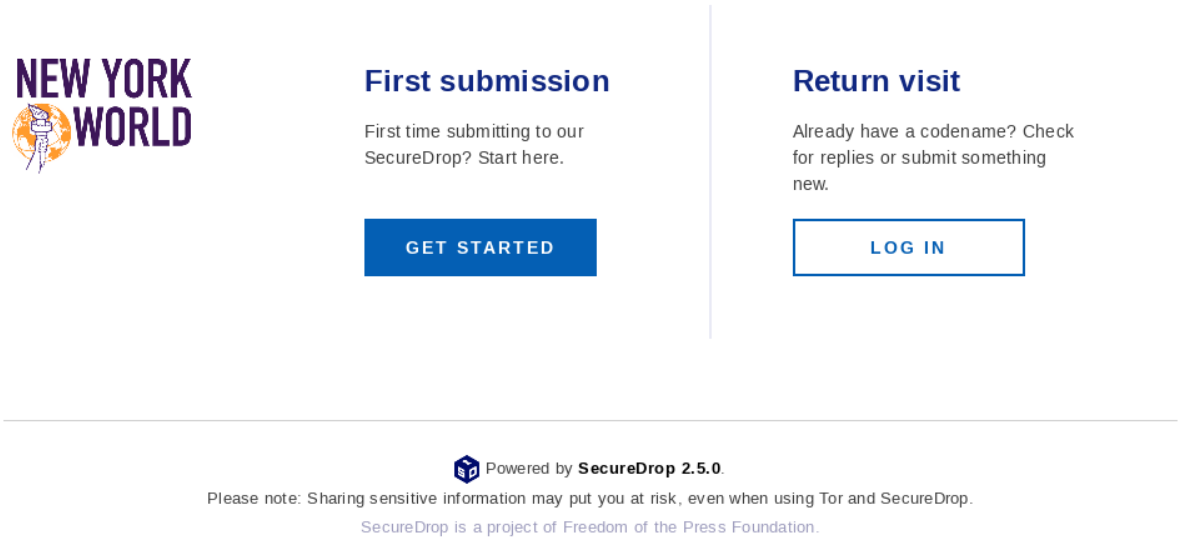
This guide provides an introduction to using SecureDrop as a *Source*. It is not exhaustive, it does not address ethical or legal dimensions of whistleblowing, and it does not speak to other methods for confidentially communicating with *Journalists*. Please proceed at your own risk. For additional background, also see the Freedom of the Press Foundation guide, How to Share Sensitive Leaks With the Press.

Warning

Freedom of the Press Foundation has no access to any other organization's SecureDrop instance, and cannot assist directly in your communications with them. If you plan to use SecureDrop to maintain your anonymity, you should not discuss your own use of it with others via unsafe methods, including email to Freedom of the Press Foundation.

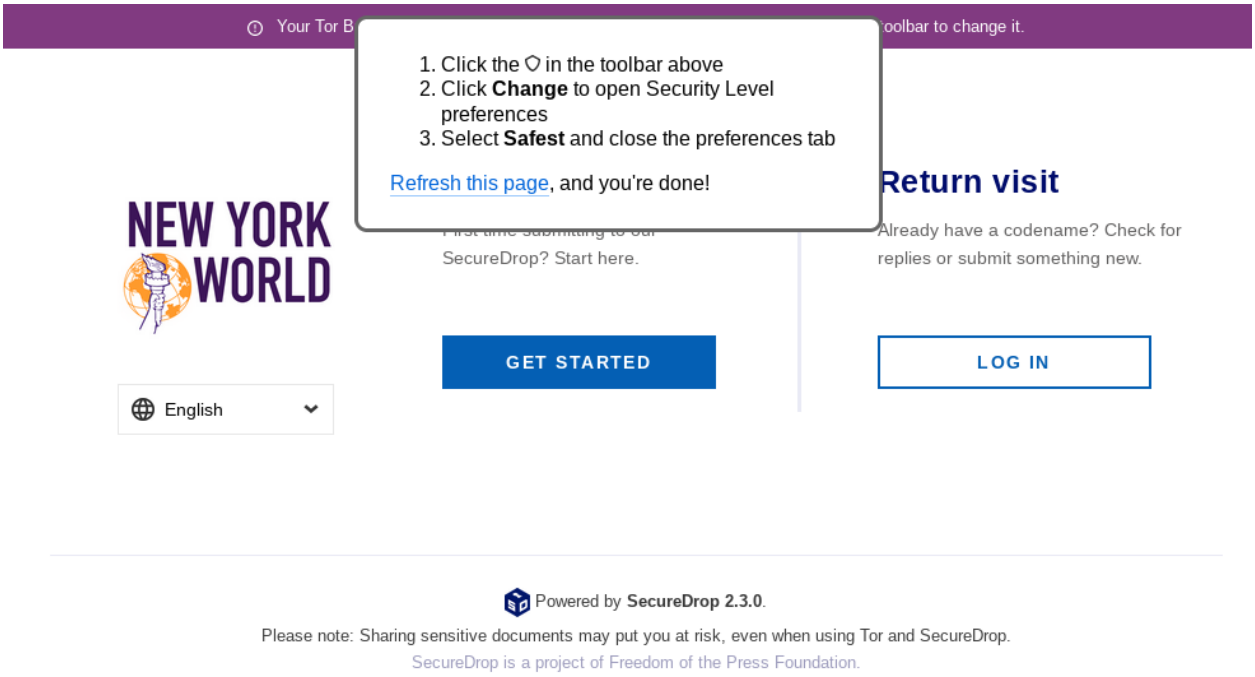
1.7.1 Making your first submission

Open Tor Browser and navigate to the onion address for the SecureDrop you wish to make a submission to. The page will invite you to get started with your first submission or to log in. It should have a logo specific to the organization you are submitting to.

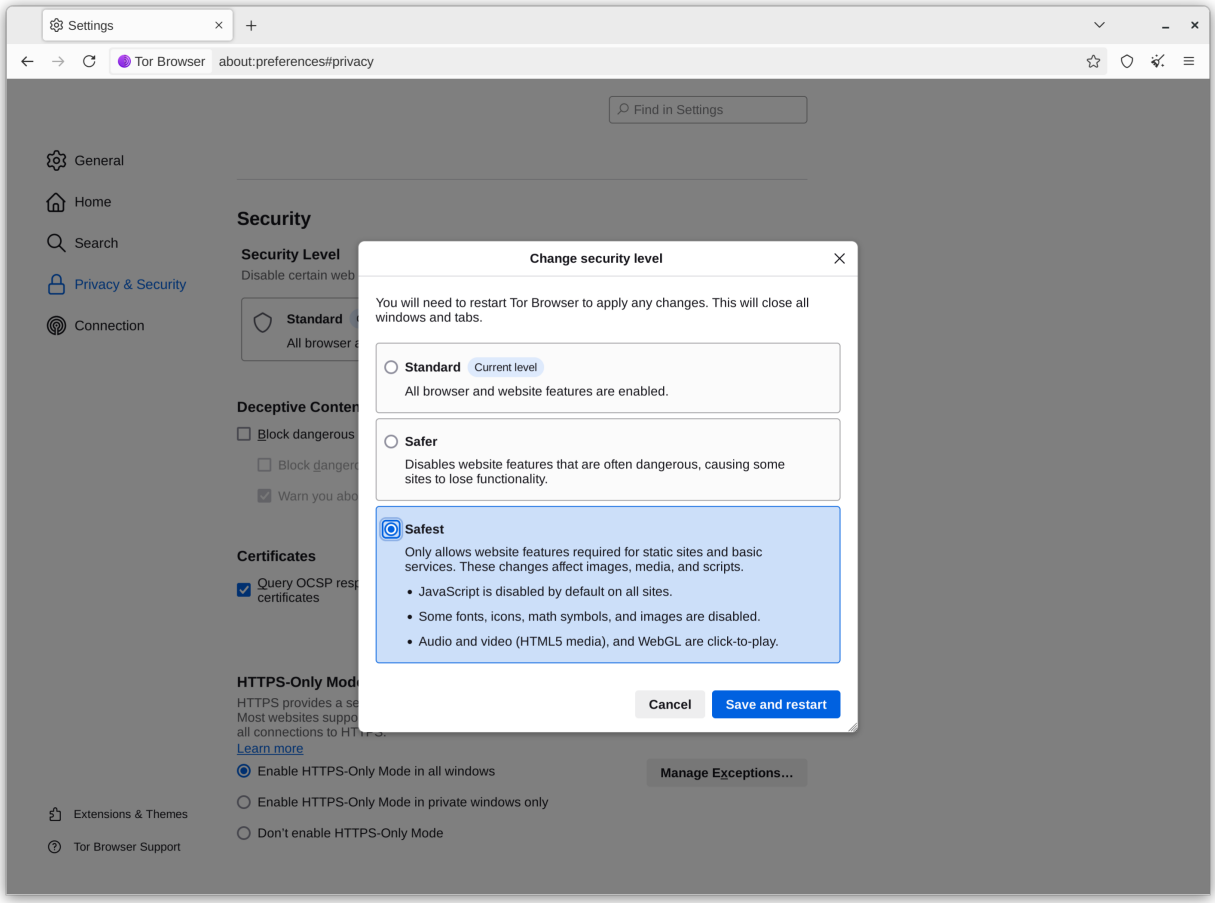


If this is the first time you're using Tor Browser, it's likely that you have JavaScript enabled and that the Tor Browser's security level is set to "Low". In this case, there will be a purple warning banner at the top of the page that encourages you to disable JavaScript and change the security level to "Safest".

Click the **Security Level** link in the warning banner, and a message bubble will pop up explaining how to increase the security level to **Safest**.



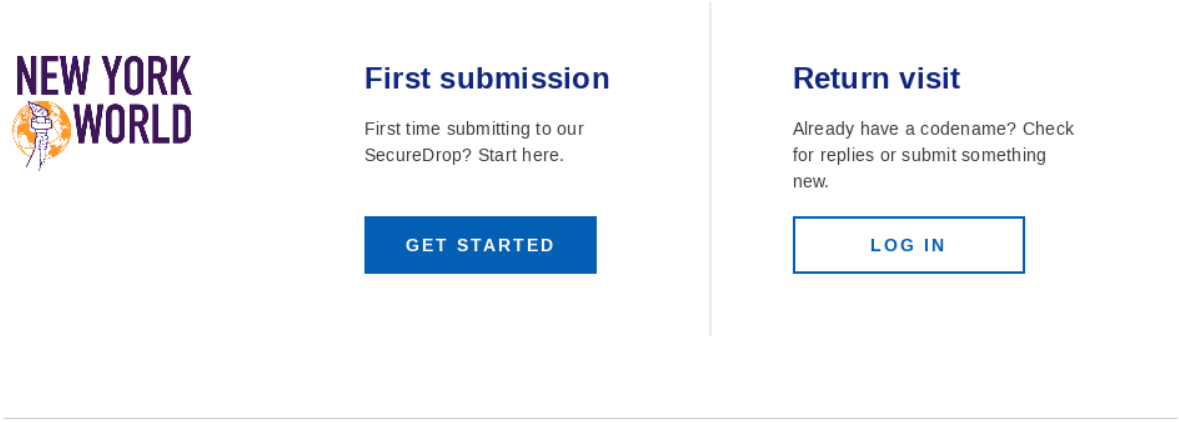
1. Click the shield icon in the toolbar
2. Click **Settings…**
3. If the current level is not already set to **Safest**, click **Change…**
4. Select **Safest**
5. Select **Save and restart** for the changes to take effect
6. Navigate back to the *Source Interface* for the SecureDrop for which you wish to submit



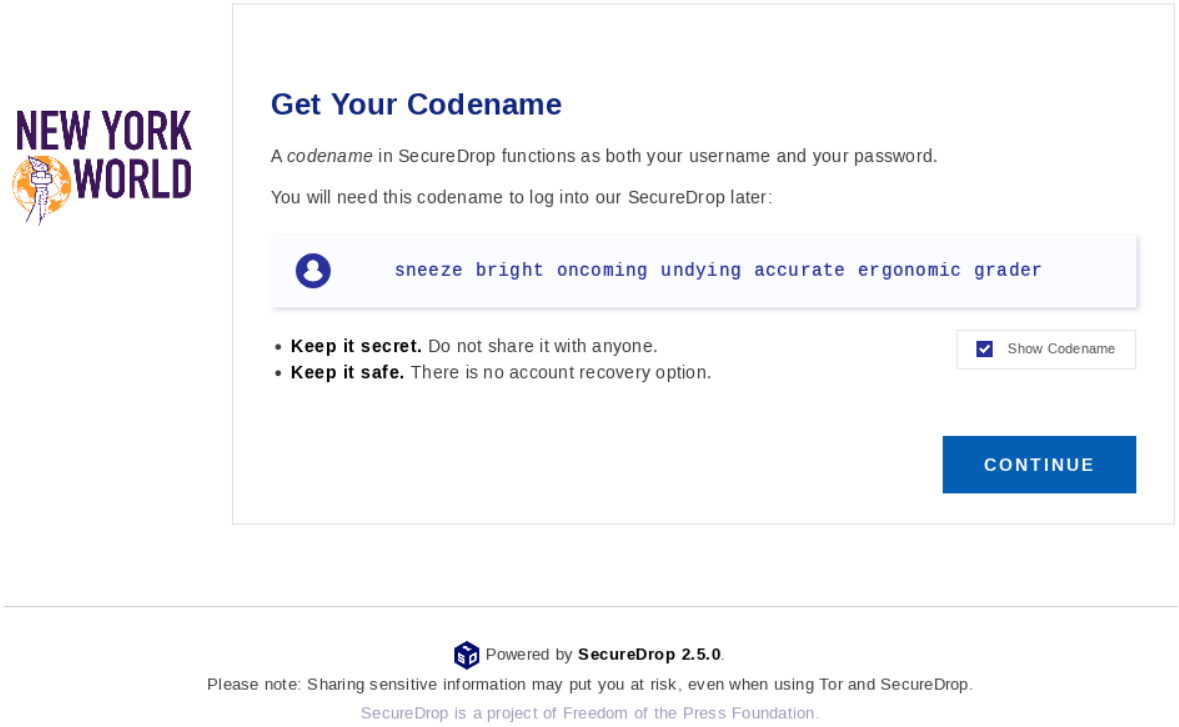
Note

The "Safest" setting disables the use of JavaScript on every page you visit using Tor Browser, even after a browser restart. This may cause other websites you visit using Tor Browser to no longer work correctly, until you adjust the Security Level again. We recommend keeping the setting at "Safest" during the entirety of the session in which you access an organization's SecureDrop instance.

Once you return to the SecureDrop page, it should stop displaying the warning. If this is the first time you are using SecureDrop, click the **Get Started** button.



You should now see a screen that shows the unique codename that SecureDrop has generated for you. Note that your codename will not be the same as the codename shown in the image below. It is extremely important that you both remember this code and keep it secret. After submitting documents, you will need to provide this code to log back in and check for responses.



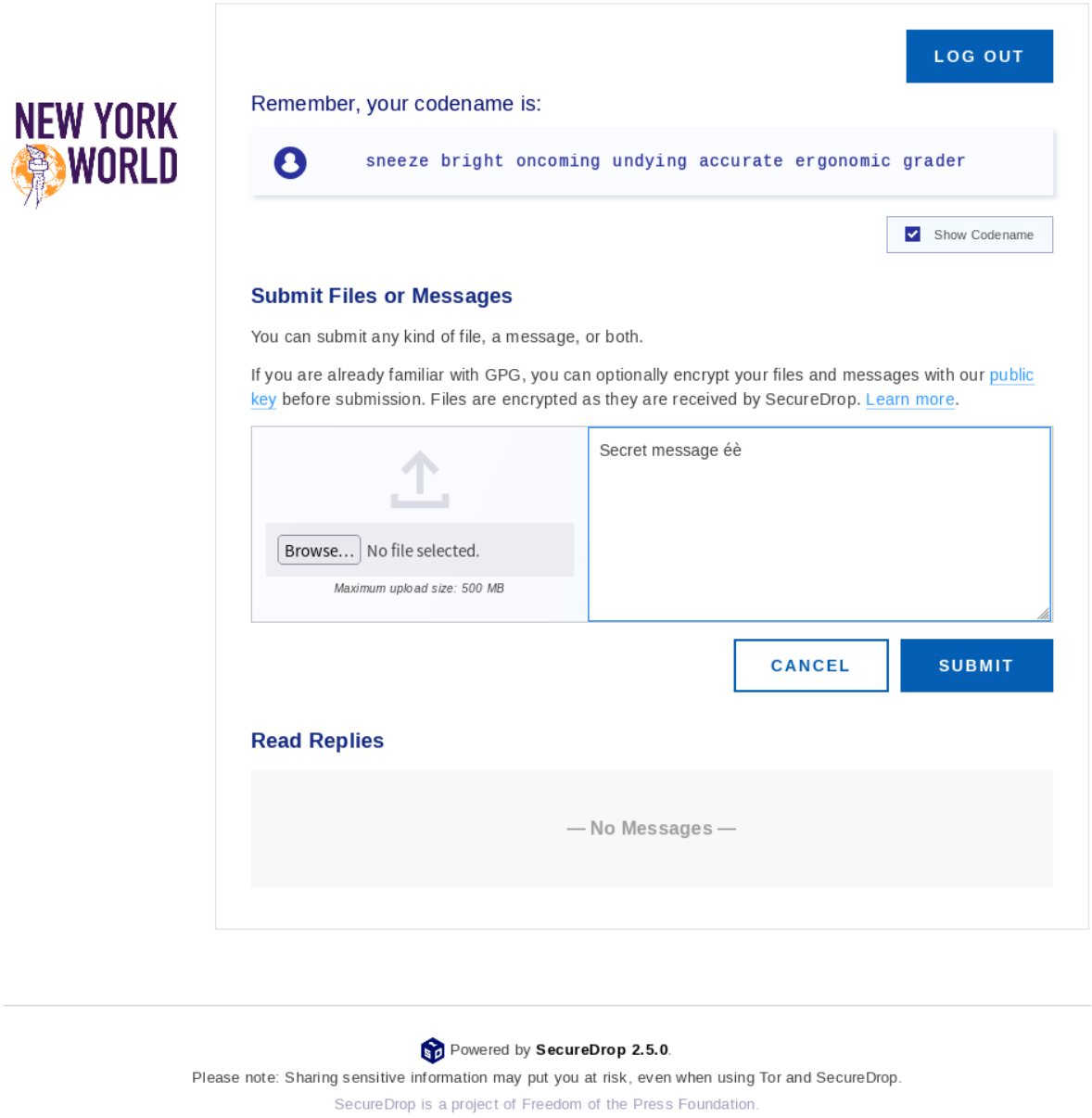
The best way to protect your codename is to memorize it. If you cannot memorize it right away, we recommend writing it down and keeping it in a safe place at first, and gradually working to memorize it over time. Once you have memorized it, you should destroy the written copy.

Your codename is associated with your pseudonymous account and all of your activity on the SecureDrop server. In

order to preserve your anonymity, you should avoid creating physical or digital associations between yourself and your codename as much as possible.

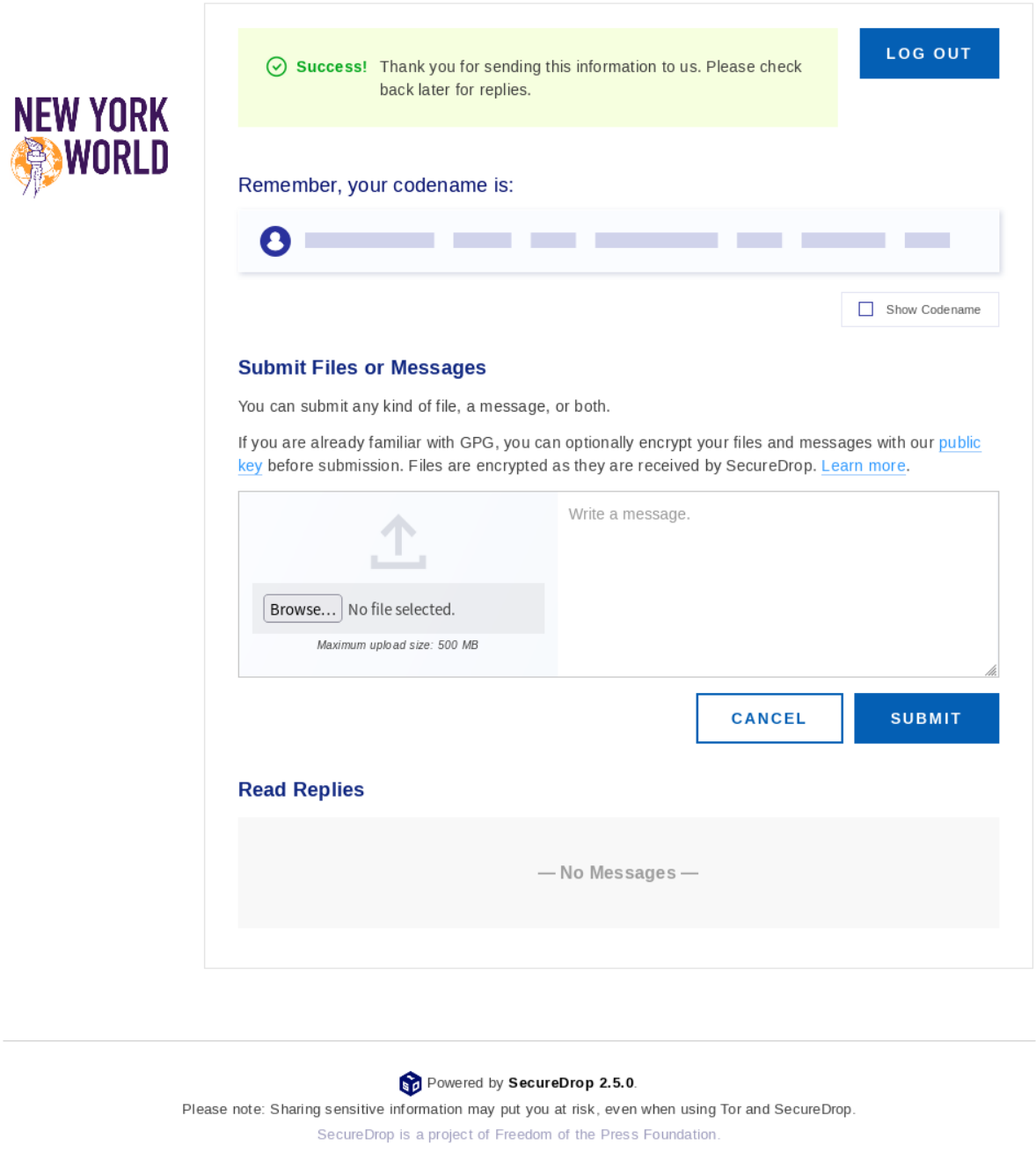
Once you have generated a codename and put it somewhere safe, click **Submit Documents**.

You will next be brought to the submission page, where you may upload a document, enter a message to send to *Journalists*, or both. You can only submit one document at a time, so you may want to combine several files into a ZIP archive if necessary. The maximum submission size is currently 500MB. If the files you wish to upload are over that limit, we recommend that you send a message to the *Journalist* explaining this, so that they can set up another method for transferring the documents.



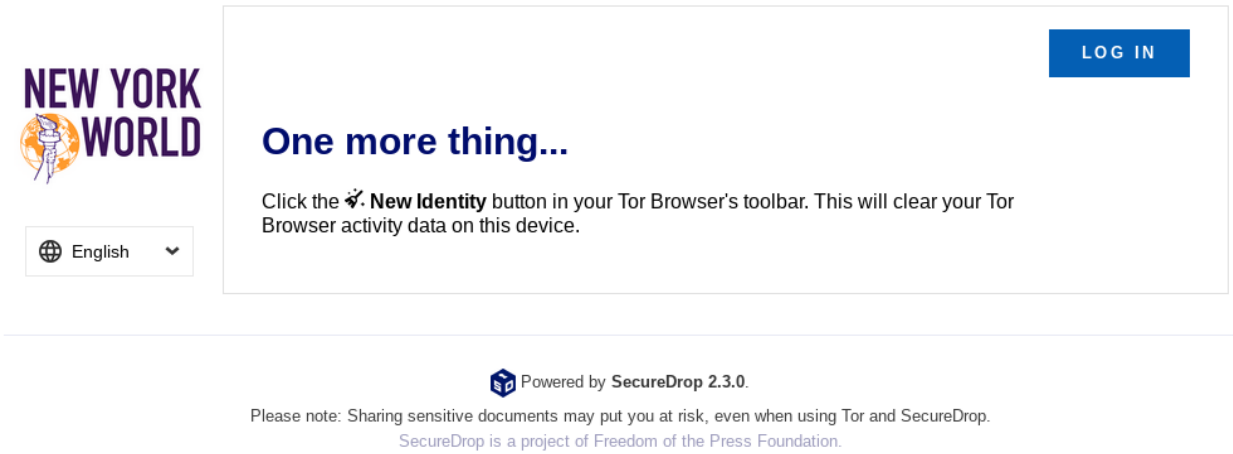
When your submission is ready, click **Submit**.

After clicking **Submit**, a confirmation page should appear, showing that your message and/or documents have been sent successfully. On this page you can make another submission or view responses to your previous messages.



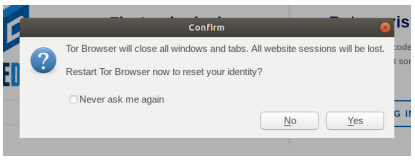
Once you are finished submitting documents, be certain you have saved your secret codename and then click the **Log out** button.

The final step to clearing your session is to restart Tor Browser for optimal security. After logging out, you should see a new page recommending you to click the **New Identity** button in the Tor Browser toolbar.



You can either close the browser entirely or follow the instructions on the page:

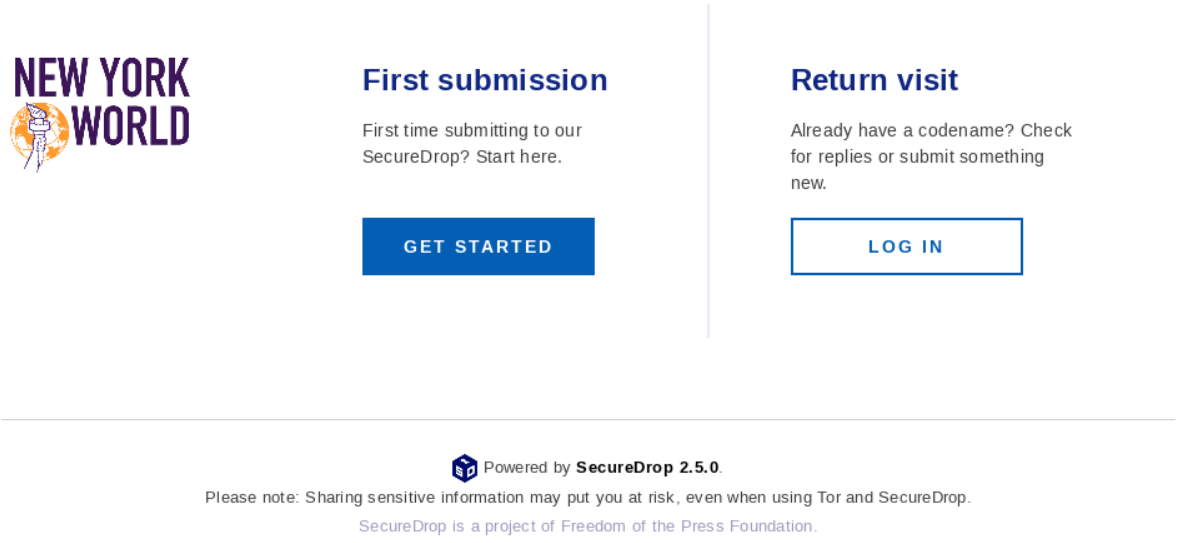
1. Click on the **New Identity** button in the Tor Browser toolbar
2. Click **Yes** in the dialog box that appears to confirm you'd like to restart Tor Browser



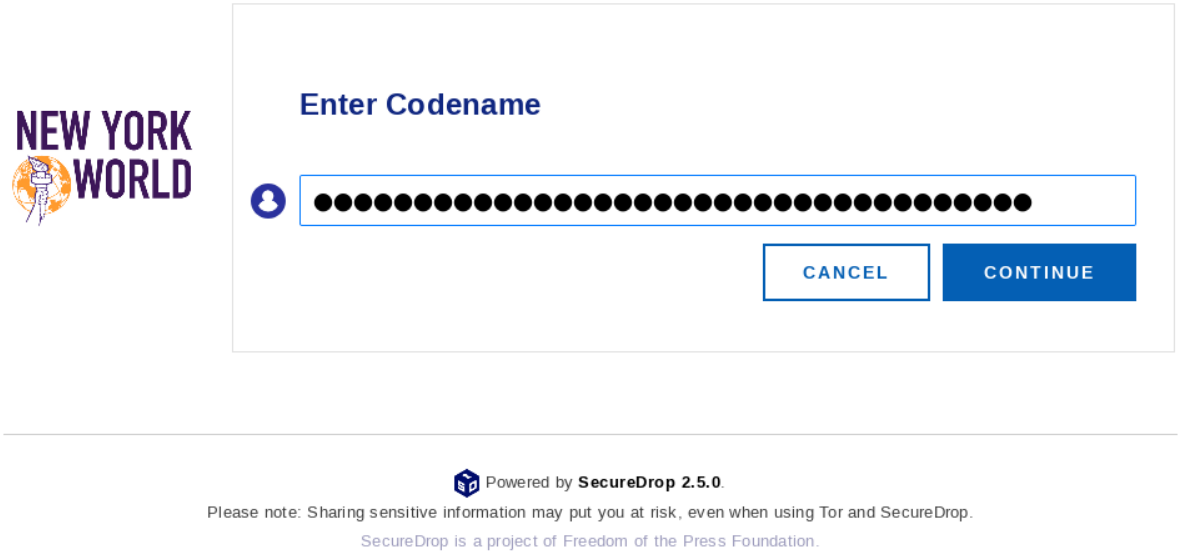
1.8 After you submit

1.8.1 Continuing the conversation

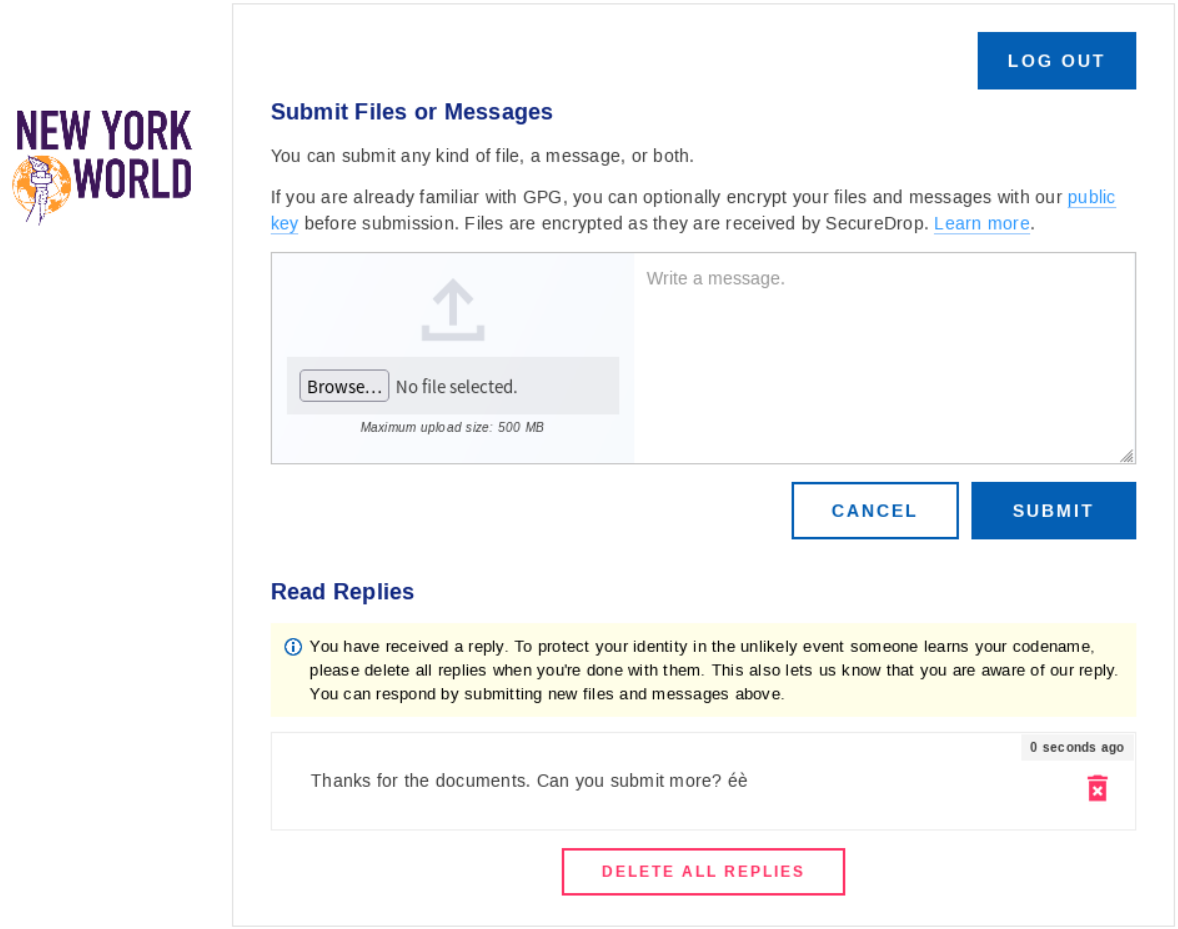
If you have already submitted a document and would like to check for responses, click the **Log in** button on the media organization's SecureDrop page.



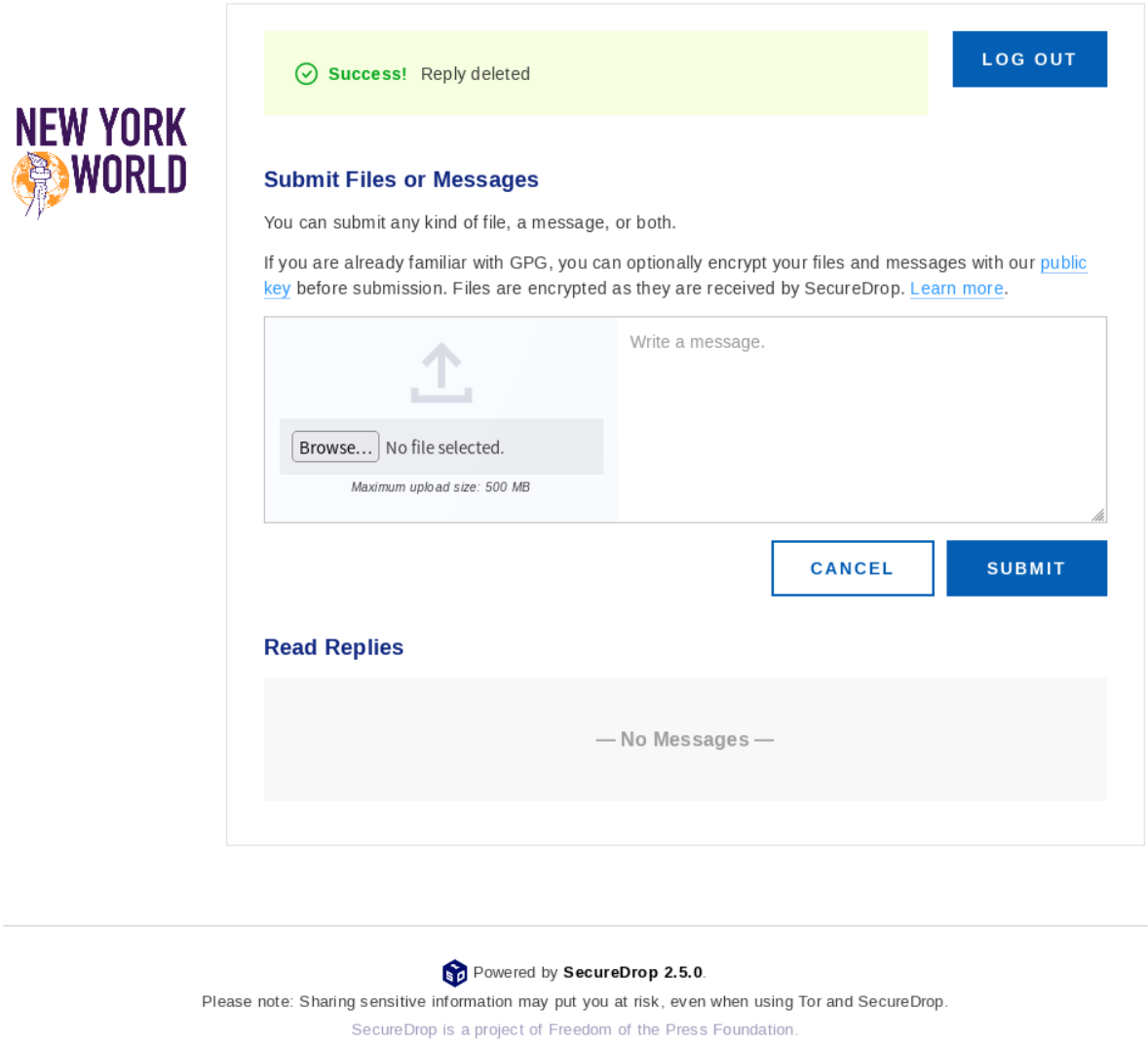
The next page will ask for your secret codename. Enter it and click **Continue**.



If a *Journalist* has responded, their message will appear on the next page. Before leaving the page, you should delete any replies. In the unlikely event that someone learns your codename, this will ensure that they will not be able to see the previous correspondences you had with *Journalists*.



After you delete the reply from the *Journalist*, make sure you see the confirmation message: “Reply deleted”.



1.9 SecureDrop for *Journalists*

Note

SecureDrop wants your feedback! Confused by something in our documentation? Let us know by opening an issue on GitHub or in our Gitter channel.

This guide presents an overview of the SecureDrop system for a *Journalist*. It covers the core functions necessary to start working with the platform: logging in securely, viewing documents, editing documents, and interacting with *Sources*.

Journalists will use the *Journalist Workstation* to read, print, and otherwise prepare documents for publication. Apart from those deliberately published, decrypted documents are never opened in an environment with direct access to the Internet

SecureDrop provides a number of benefits intended to protect *Journalists*. Communications through SecureDrop are encrypted in transit, so messages cannot be easily intercepted and read while they are moving across the Internet, and

are also encrypted on the server so if any attacker manages to break into the server, they would not be able to read past messages.

In addition, the decryption key for submissions resides in an air-gapped environment (not connected to the Internet), which makes it harder for an attacker to access.

It also helps in the event of a subpoena or court order. All servers are owned by the individual news organization, so no third-party companies can be secretly subpoenaed. Additionally, SecureDrop limits the amount of metadata it collects and saves, so there's no trail showing exactly when a *Journalist* was exchanging messages with a *Source*, or details that might give the *Source* away.

For full details about what makes SecureDrop a unique and useful tool for *Journalists*, *see here.*

1.10 Starting Qubes

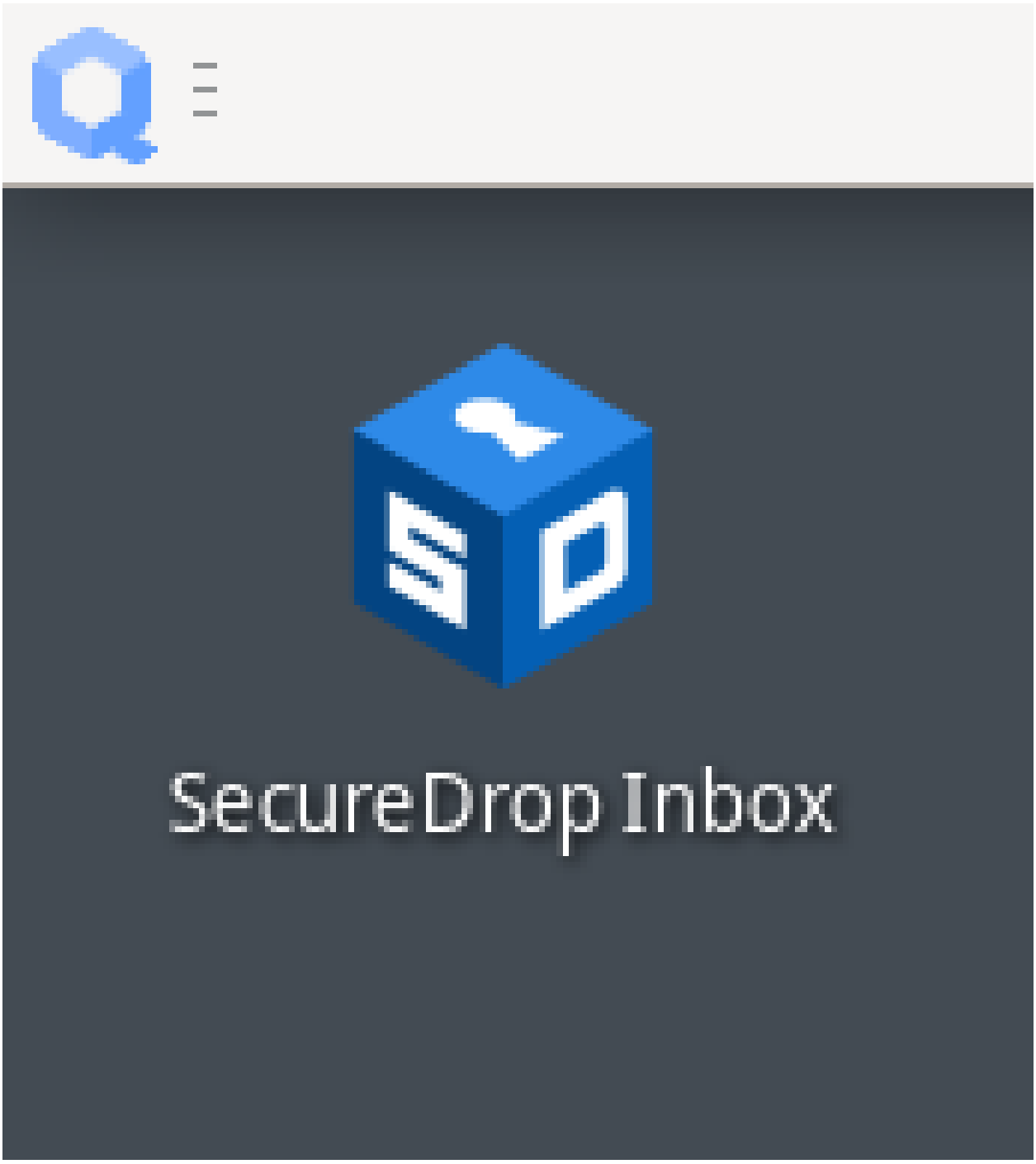
When turning on the *Journalist Workstation*, you will be greeted with a password prompt. This is the full-disk encryption passphrase.

This passphrase protects your entire system. It is of the utmost importance to secure this passphrase. When not using the *Journalist Workstation*, shut down the computer completely so as to take advantage of the protections offered by full-disk encryption.

After entering the passphrase, Qubes OS will boot. Log in with the username and password set up by your administrator.

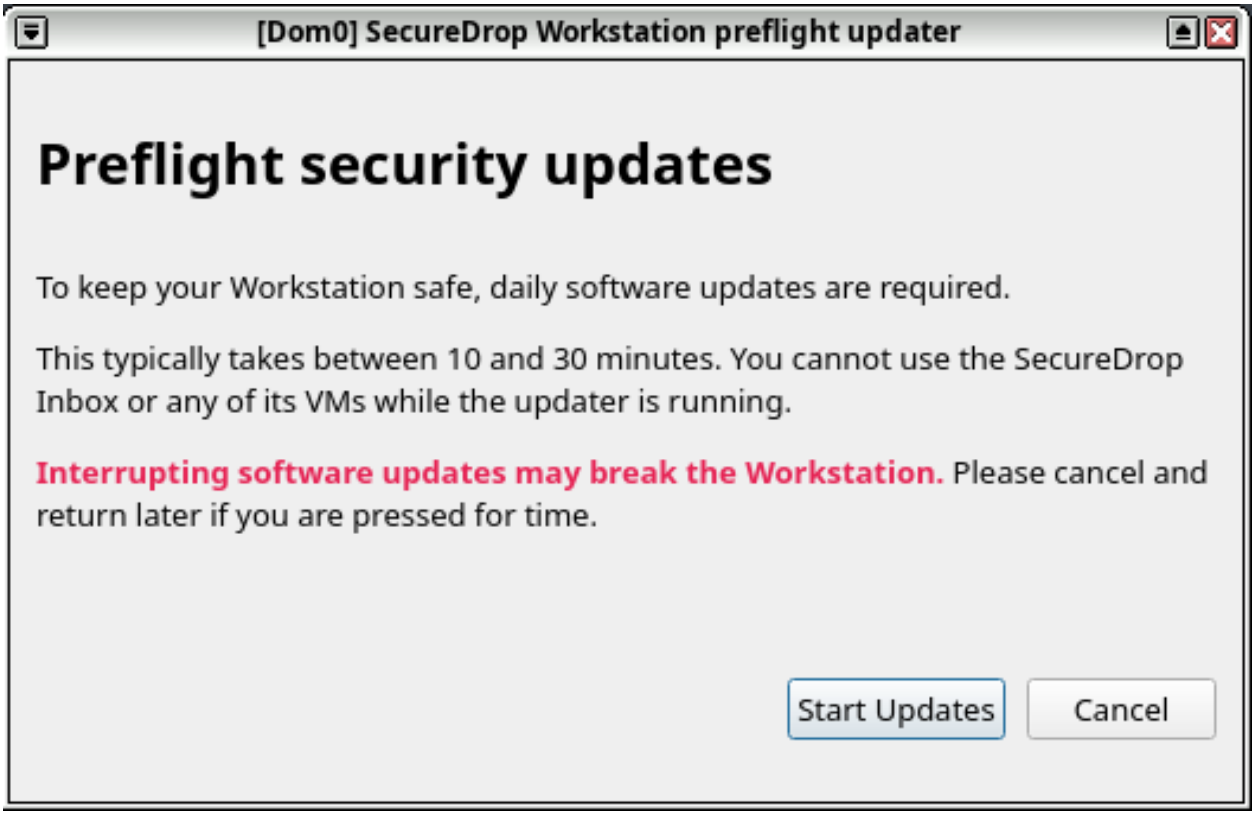
1.11 Starting SecureDrop Inbox

After you log into Qubes, SecureDrop Inbox will start automatically. If you have previously exited the application, you can double-click on the **SecureDrop** desktop shortcut to launch it.



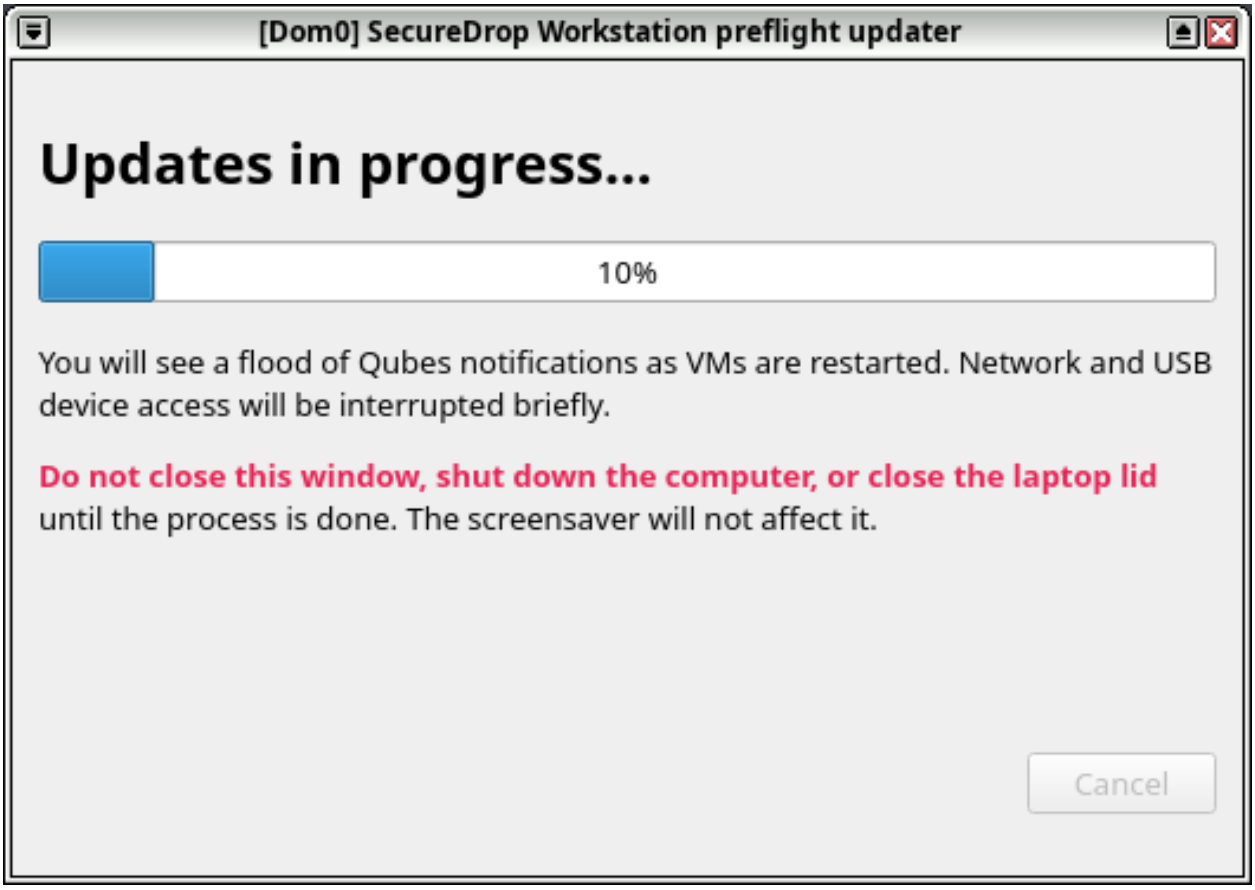
1.11.1 Performing updates

Unless the system has just been updated, you will now be prompted you to automatically download and apply any available security updates:



For security reasons, you will not be able to launch SecureDrop Inbox until updates have been applied. This typically takes between 10 and 30 minutes.

Click "Start updates" if you are ready to start the process. (If you prefer to shut down the machine or do other work in Qubes OS instead, click "Cancel".) You will see a progress indicator until updates are completed:



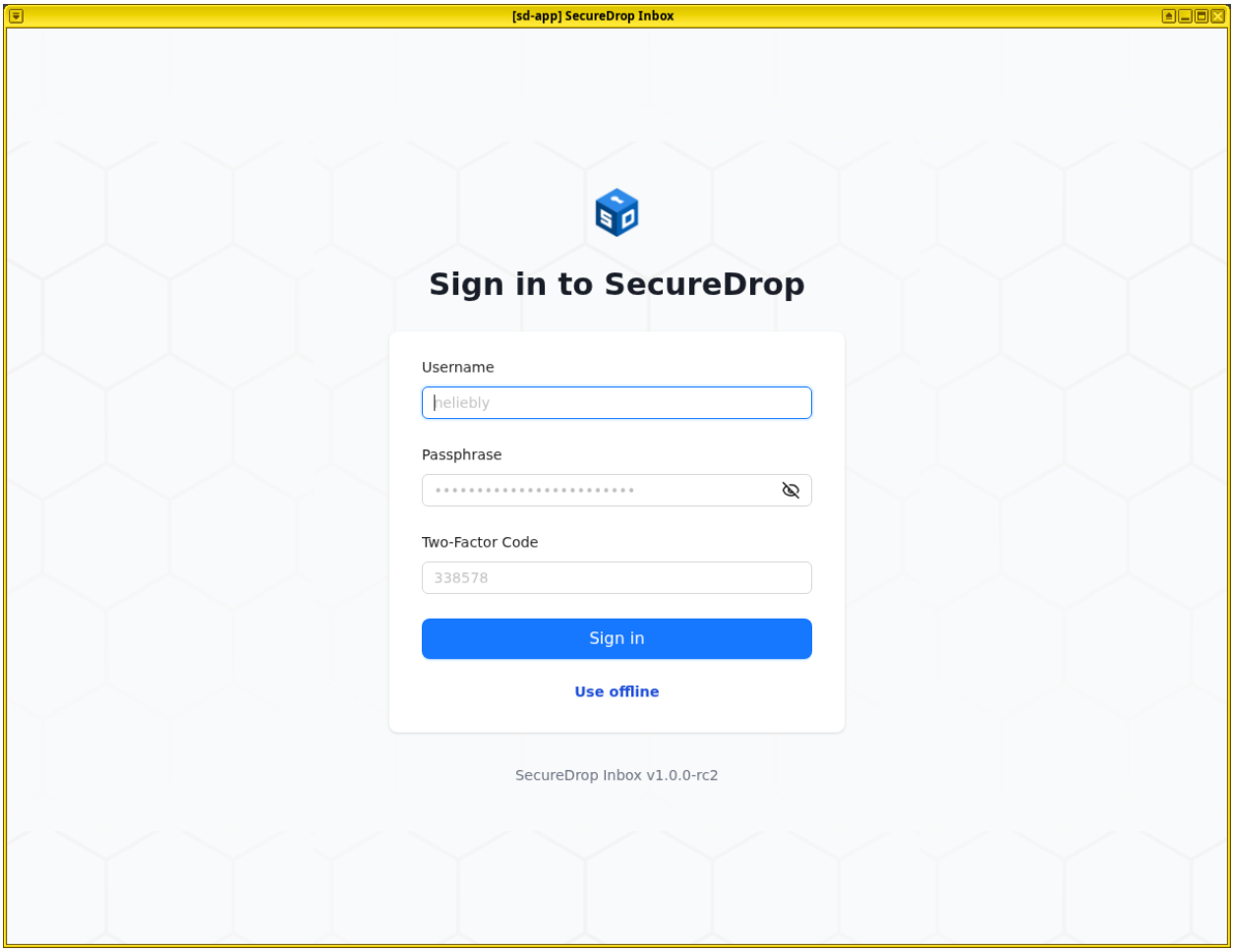
Important

Allow the update process to complete fully, without closing or interrupting it, or you risk breaking important system components.

At the end of this process, you may be prompted you to reboot if core system components were updated. Once all steps in the update process have been completed, SecureDrop Inbox will launch automatically.

1.11.2 Signing in

To sign in, enter the username and passphrase provided to you by your SecureDrop administrator, as well as the two-factor code using the method you have set up. If you have used SecureDrop Workstation before, these are the same credentials that you would use to log in to the *Journalist Interface*.



Troubleshooting tips

If you have trouble running the updater or logging in, please contact your administrator. Our *network troubleshooting guide* for administrators gives detailed steps for investigating connectivity issues.

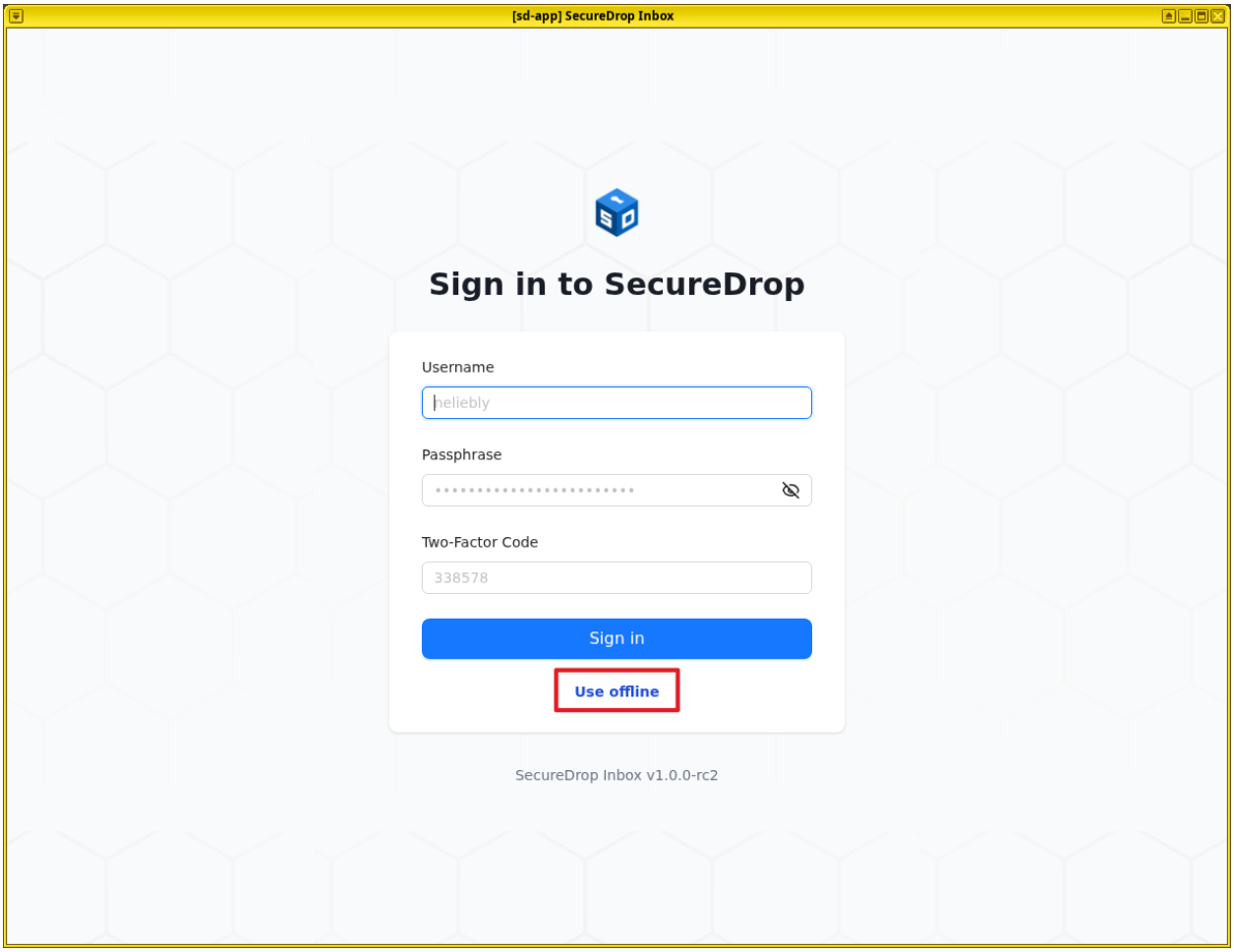
1.11.3 Seen and unseen submissions

Sources with submissions (messages or files) that have not been seen by any *Journalist* will be displayed in bold text in the source list.

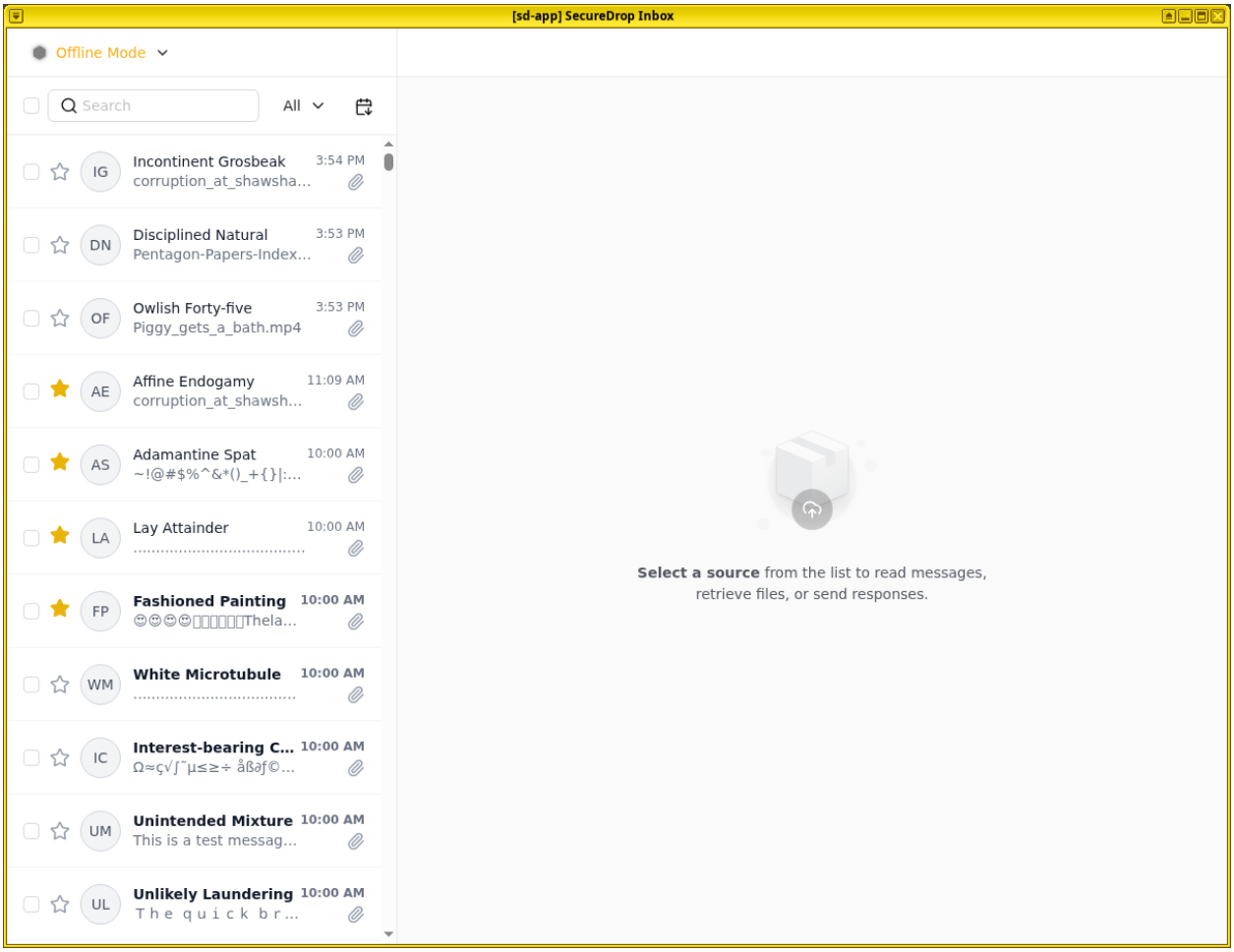
As soon as any *Journalist* clicks on a source with unseen submissions, it will be marked as seen (no longer displayed in bold text) for all users.

1.11.4 Working offline

Offline mode is available for circumstances where you wish to work offline or are unable to connect to the SecureDrop servers. In offline mode, any content that you have previously downloaded will be available. You will not be able to send or delete messages, and your actions will not impact the seen/unseen state of submissions.



Because SecureDrop allows you to download and decrypt submissions on one machine, submissions that you have downloaded are still available in offline mode and can be accessed even when you are not logged in.



Important

Protecting downloaded submissions is another reason why The *Journalist Workstation* needs to be powered off completely when it is not in use.

1.12 Communicating with sources

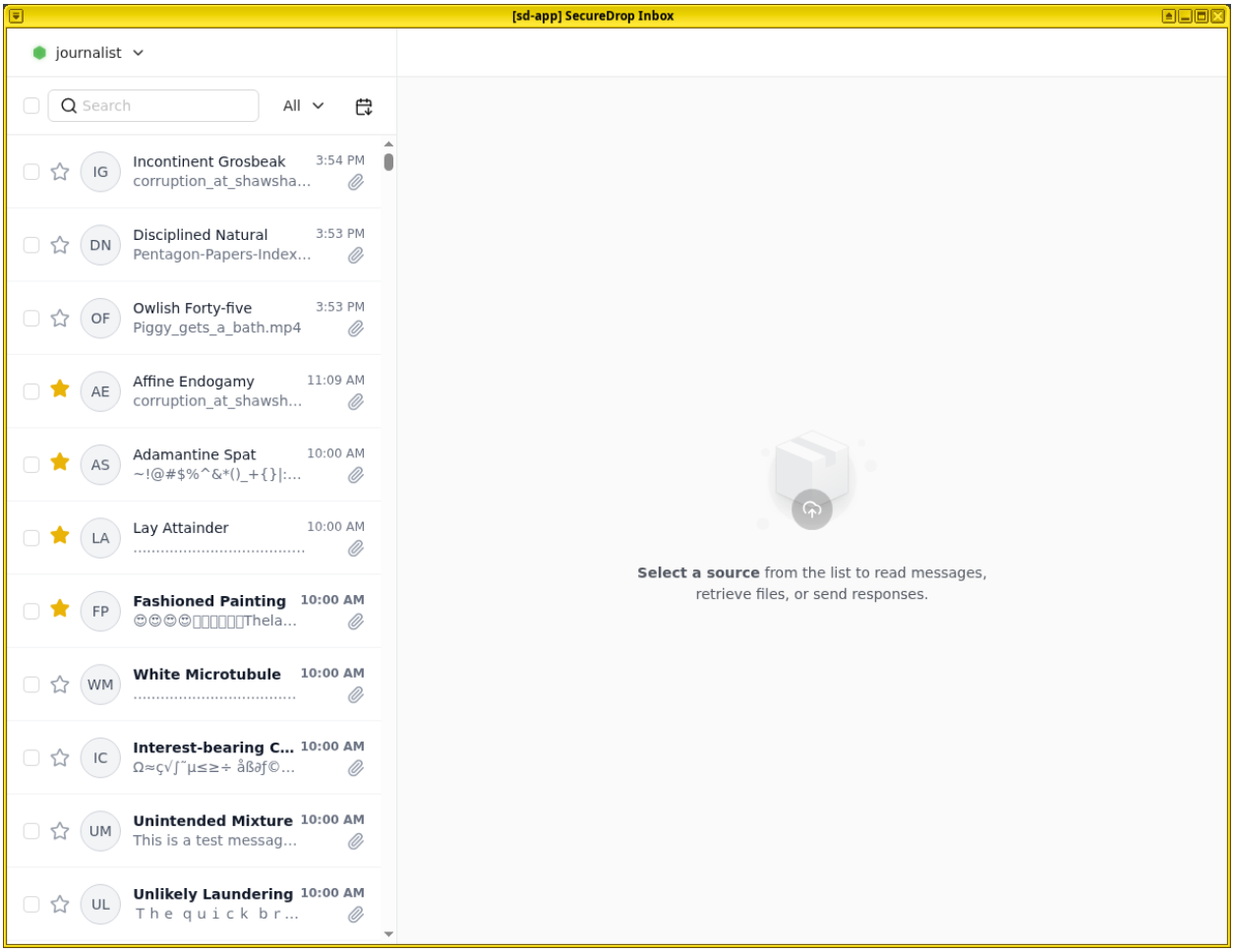
The *Journalist Workstation* lets *Journalists* check SecureDrop, decrypt and securely view submissions, and reply to sources, all on the same computer.

Once logged in, you will see a chat-like user interface:

- The top of the left panel shows your username, if you are logged in, or the sign-in button.
- The action area of the left panel provides the ability to search for sources, toggle the sort order, select multiple sources, and delete sources.
- The larger portion of the left panel holds the list of sources that have submitted to your instance. Each source is identified to you with a two word pseudonym. You will also see the date of the last source activity, an icon to indicate if a source contains attachments, and a button to mark a source as starred.
- The right panel holds the conversation view. All parts of the conversation with a specific source (messages, files, and journalist replies) will be displayed here.

1.12.1 Opening a conversation

To display a conversation in the conversation view, simply click a source in the source list.



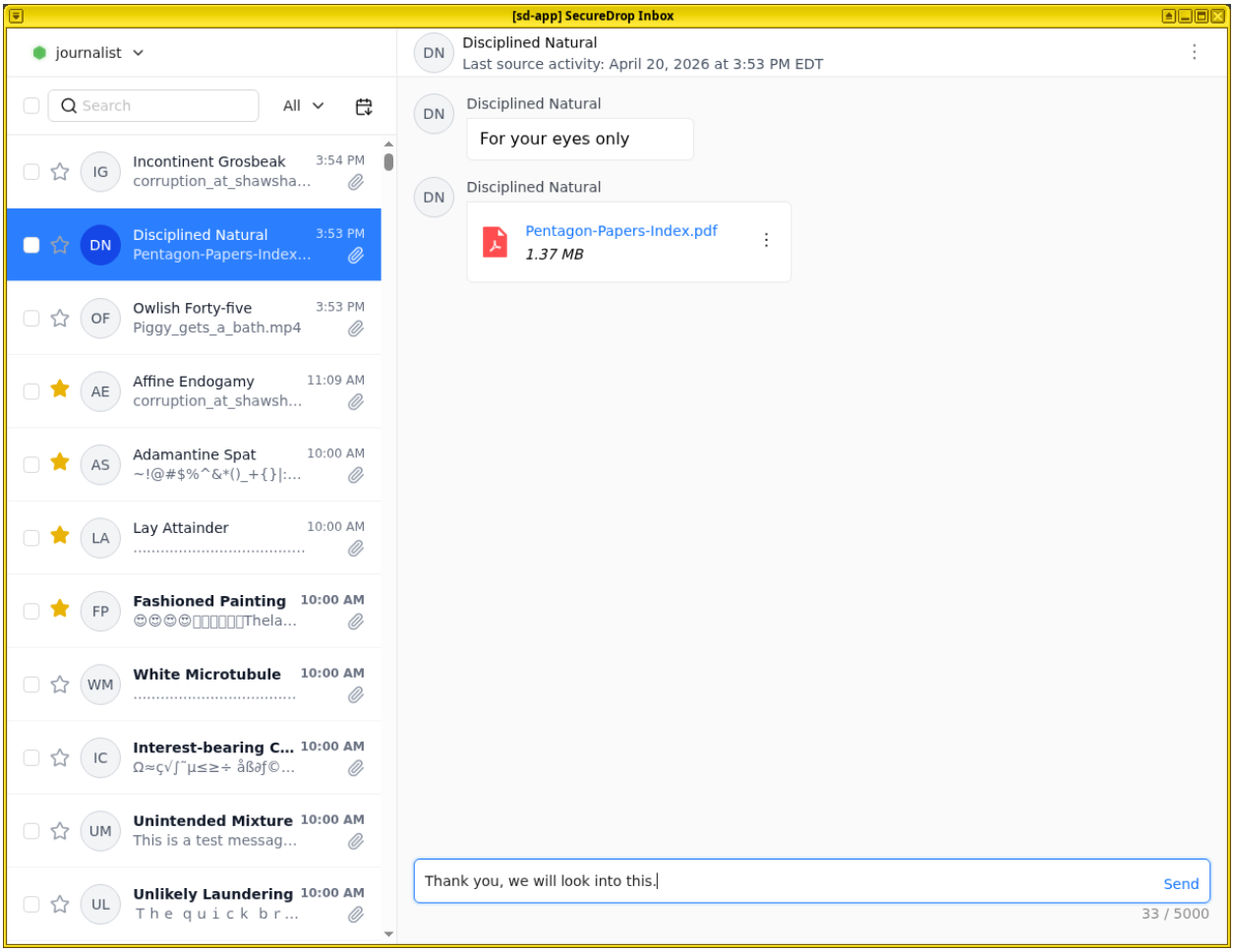
Journalists sending replies are assigned different colors and identified with their initials. Move your mouse pointer over the initials to reveal the full name.

1.12.2 Highlighting conversations

You can highlight important conversations by clicking on the star beside a source's name. Starred sources will be visible as starred to everyone in your organization.

1.12.3 Sending a reply

Compose a reply to the selected source in the text box at the bottom of the conversation view. Click the **Send** button or press "Ctrl+Enter" to send a reply. Any replies you did not send will be discarded when you move to a different conversation.



Note

If a reply fails to be sent successfully, it will still be visible in subsequent sessions, including to any other users logging into the same physical *Journalist Workstation*.

1.12.4 Deleting conversations and source accounts

As part of routine SecureDrop usage, we recommend that you establish data retention practices consistent with your organization's threat model, data lifecycle and data retention policies. Regularly deleting conversations and source accounts can mitigate risks in the event that your SecureDrop servers or a source's account details are compromised.

If you delete messages and files for a source, the source will continue to appear in the list of sources in SecureDrop Inbox, and they will still be able to log into the *Source Interface* using their codename. Consider using this option as part of regular deletion of reviewed submissions, especially if you are not sure that all communication with the source has concluded.

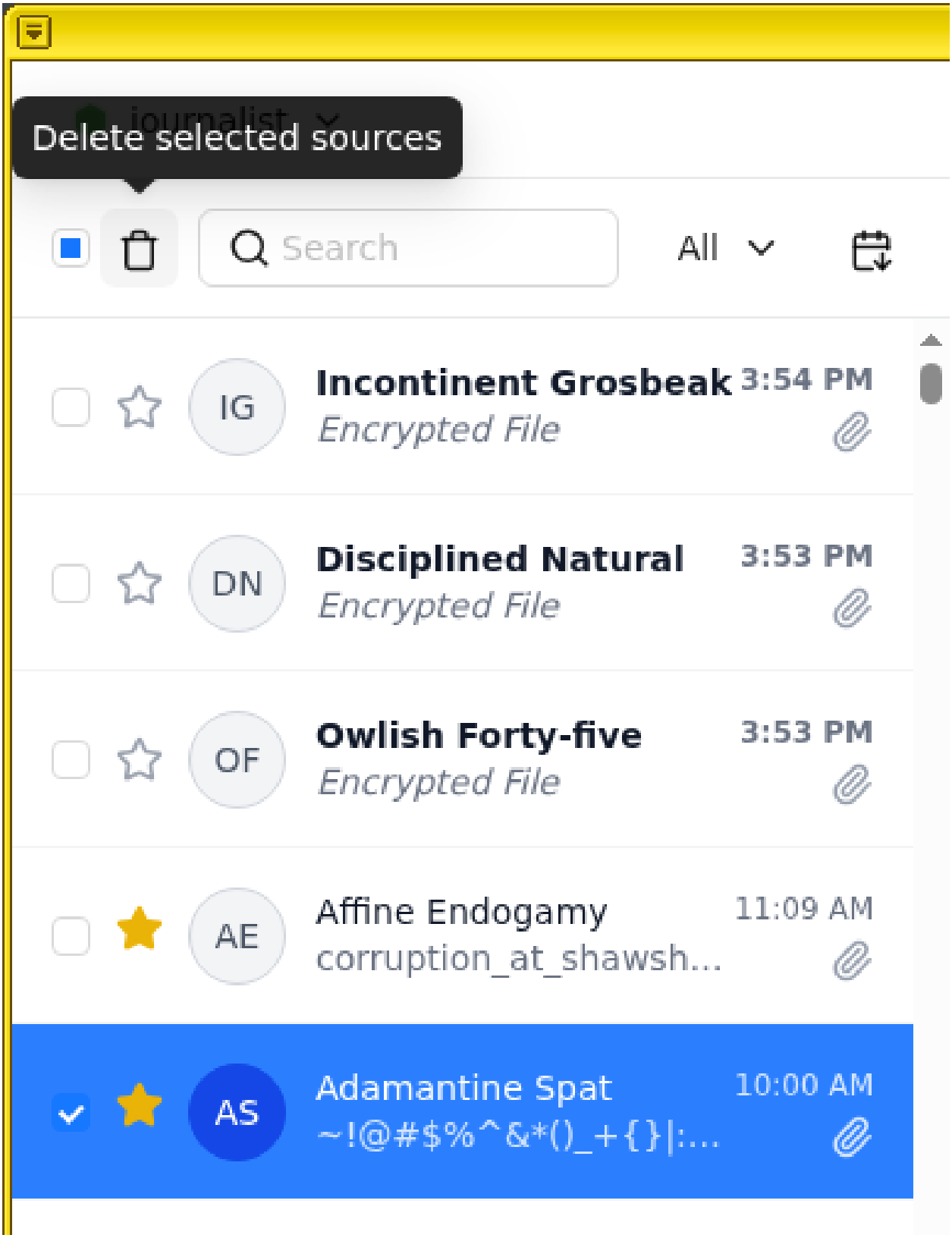
Note

If you delete all messages and files, that includes all replies you have sent to the source, even if the source has not seen them yet. You will still be able to send new replies.

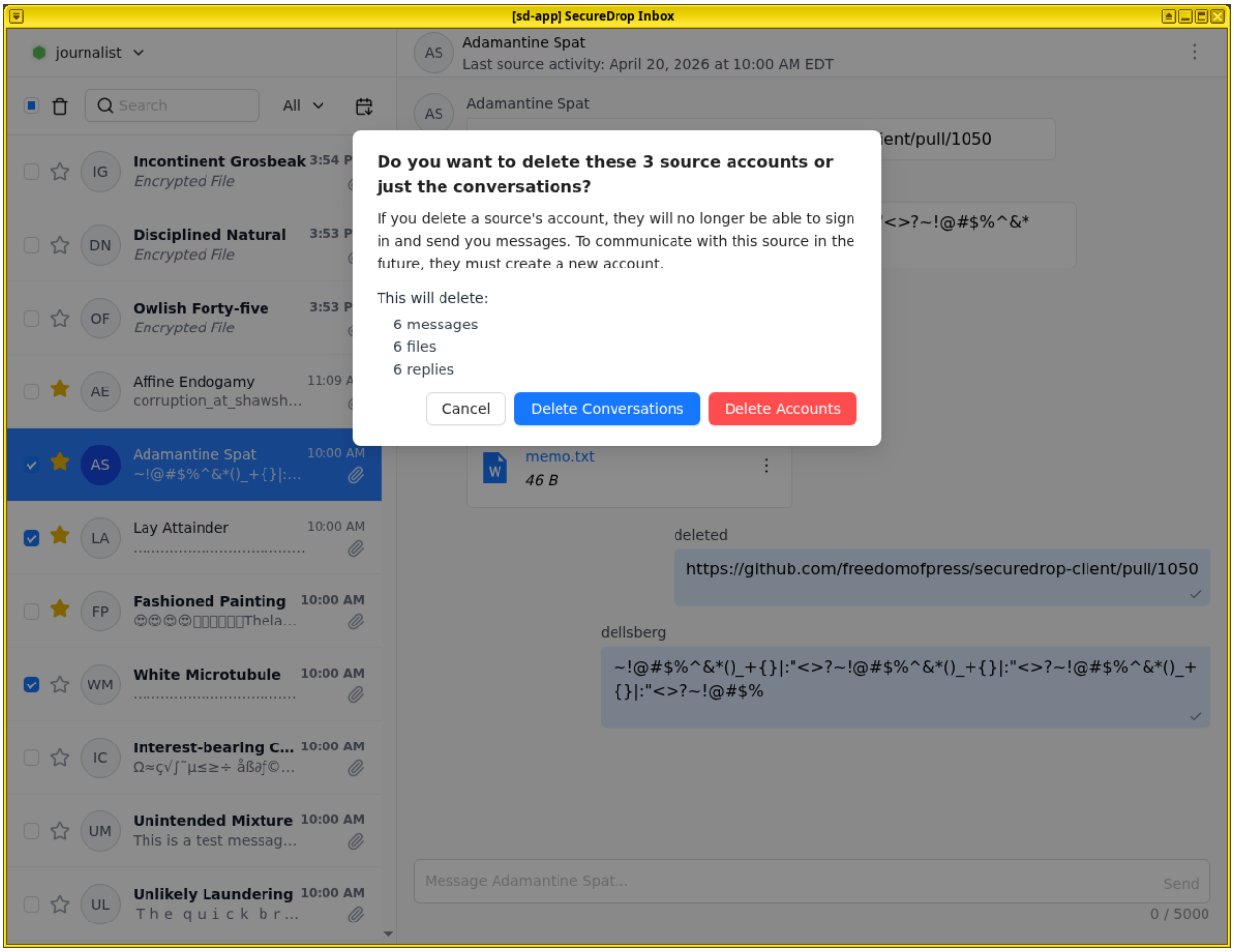
If you delete the entire source account, the source will not be able to log in again using their codename, and all information about them will be destroyed. Consider using this option if it is clear that all communication with the source has concluded, or if the source has requested that all information about them and their submissions should be removed.

Deleting conversations

You can delete a single source conversation checking the box beside the Source name in the list, then clicking the delete button (as indicated by a trash icon) in the action area at the top.



You will be presented with a pop-up where you will be asked to confirm if you would prefer to **Delete Conversation** or **Delete Account**.



Click **Delete Conversation** to delete all files and messages (including journalist replies) associated with this source, while keeping the source account active. The source will continue to appear in the source list, and will be able to communicate with you through the *Source Interface*.

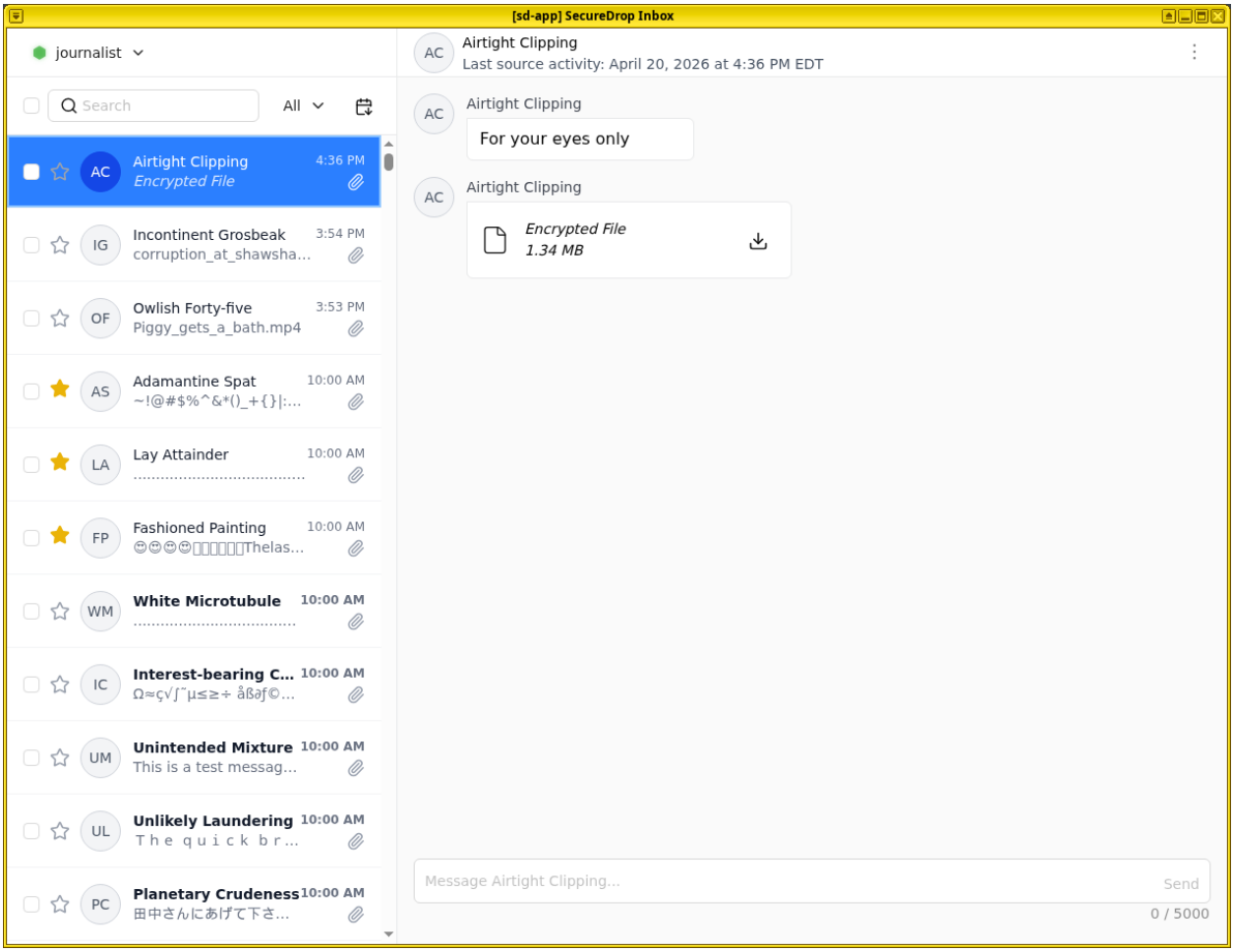
Click **Delete Account** to also remove the source from the source list, and to prevent them from logging into the *Source Interface*. Their account will be completely removed from the system.

Deleting multiple conversations

To delete multiple conversations or accounts, select more than one source conversation from the list, then click the delete button. You will be presented with the same options to **Delete Conversations** and **Delete Accounts** as you would with a single source conversation.

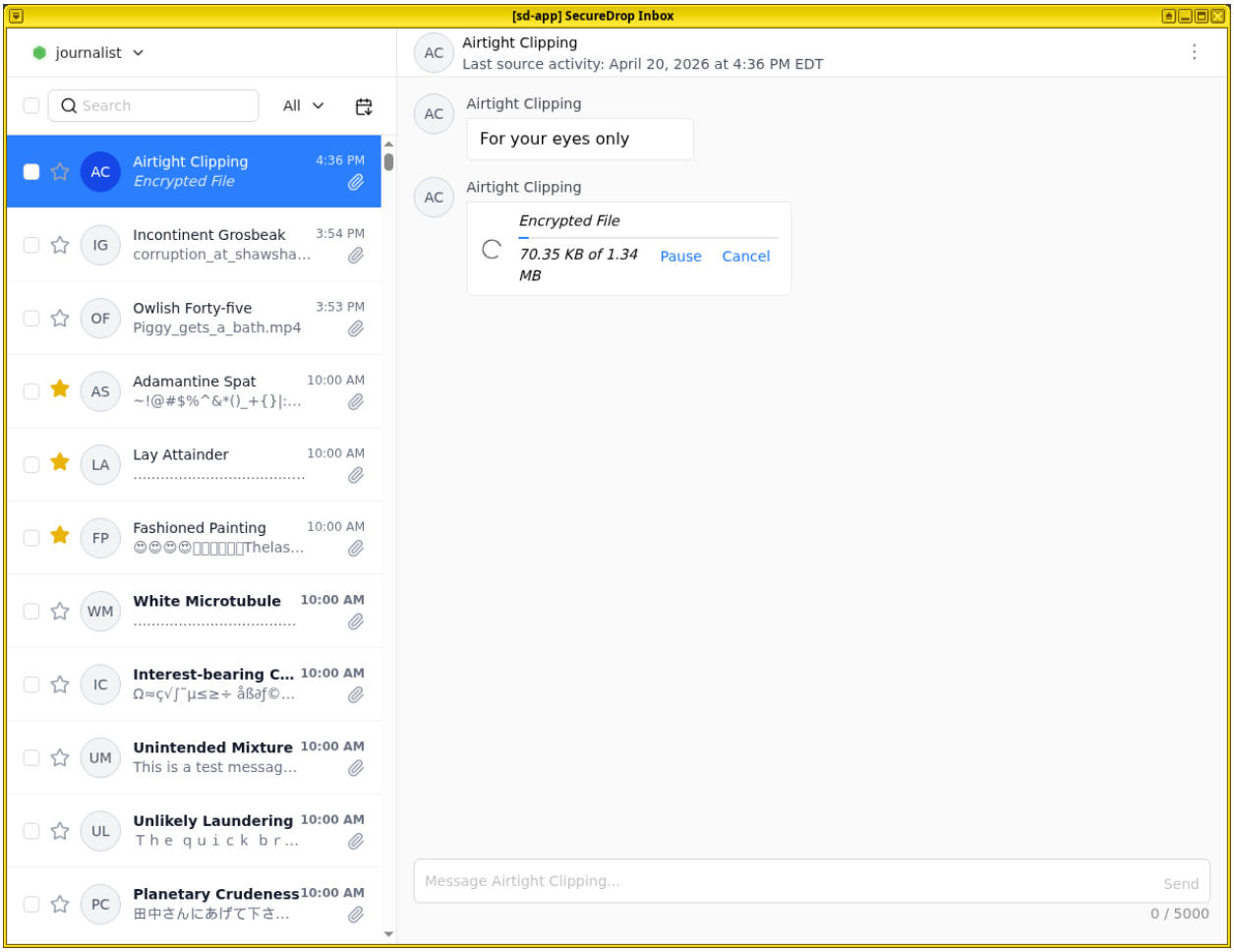
1.13 Working with submissions

When a *Source* submits files, you will see a Download button in the conversation flow, a file size, and light-gray text that says "Encrypted File."



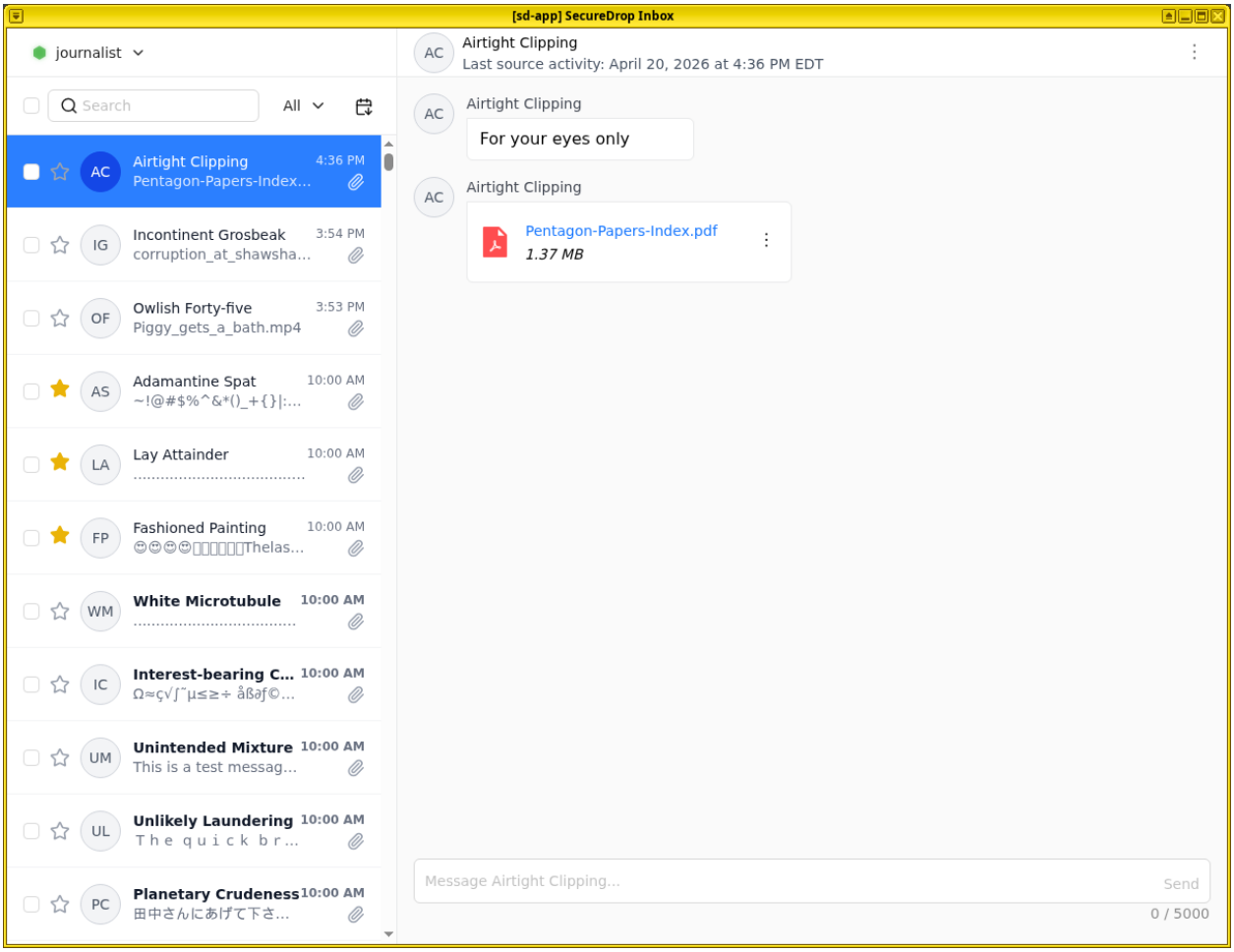
1.13.1 Downloading

To download a file, click the **Download** button. An animated spinner will indicate that the file is downloading, and a progress bar will indicate the download's progress:



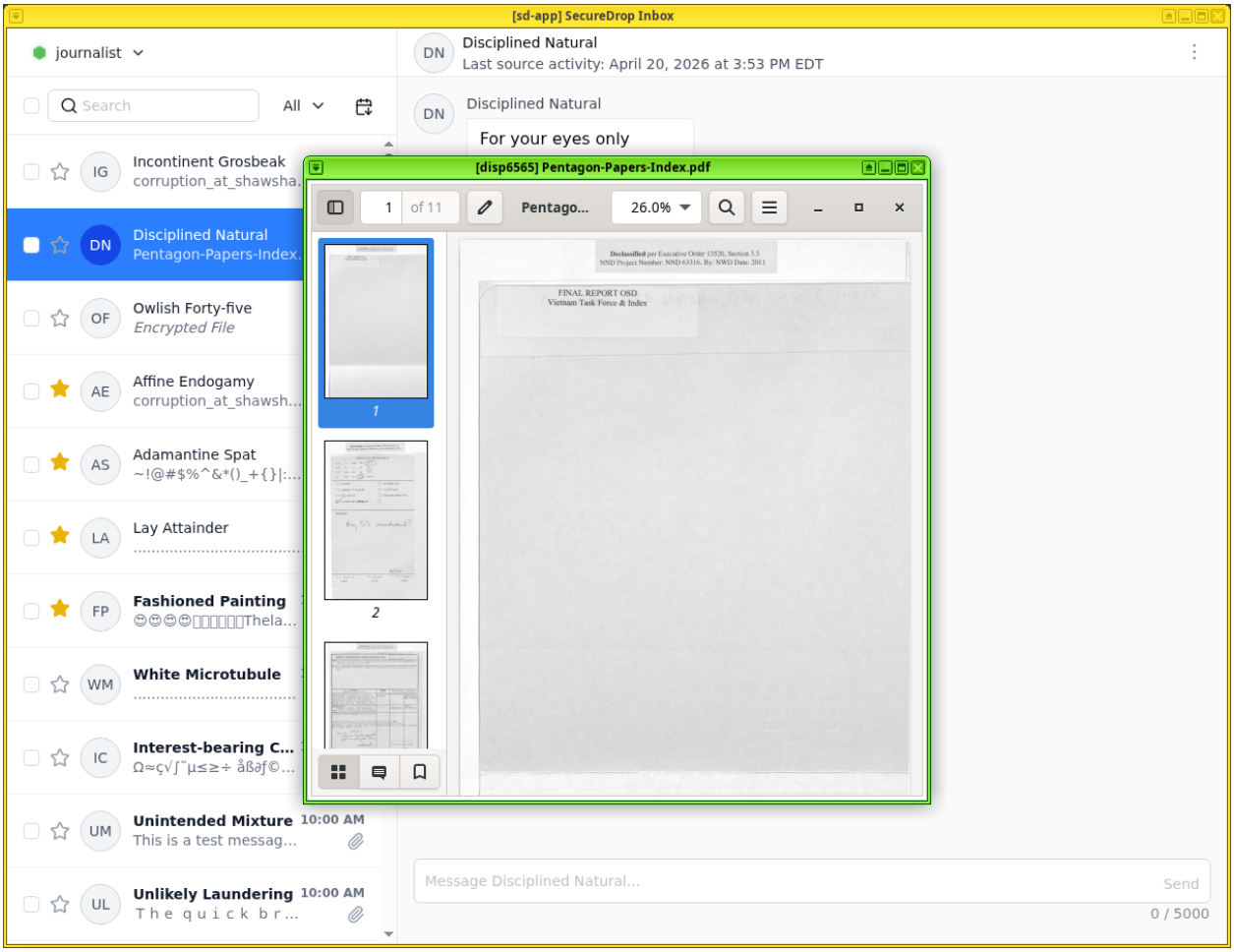
If necessary, you can pause the download by clicking “Pause,” and resume the download later with “Resume.”

Once the file has been downloaded and decrypted, the filename will be visible, as will the action **Export** and **Print**. The displayed file size may increase after the download is complete, because the SecureDrop Client automatically decompresses the downloaded file.



1.13.2 Viewing submissions on the *Journalist Workstation*

To view a downloaded submission, click its filename. This will open the file in a temporary environment, called a "disposable VM." The file you clicked on will open in a new window with a different colored border and a window title prefixed with "disp" (meaning disposable).



This disposable VM is a special isolated environment; it does not have internet access, and isolates the files that you are viewing from other sensitive files and applications on the *Journalist Workstation*.

Supported filetypes

The following filetypes are currently supported for viewing on the *Journalist Workstation*:

- .txt, .csv, .pdf
- Microsoft Office files (.doc, .docx, .xls, .xlsx, .ppt, .pptx)
- OpenDocument files (.odt, .ods, .odp)
- Audio: .mp3, .mp4, .mpeg, .wav, .ogg (Ogg Vorbis)
- Video: .mp4, .webm, .mov (Quicktime), .avi (Audio Video Interleave - Microsoft), .wmv (Windows Media Video)
- Image: .gif, .png, .jpeg, .tiff, .svg, .ico, .webp, .heic, .avif
- Compressed archives: .zip, .tar.gz (although printer support for files inside an archive is still to be implemented)

A full list of supported filetypes can be found here.

Tip

In Qubes, window border colors are used to signify different virtual machines.

1.13.3 Printing submissions from the *Journalist Workstation*

To print a document, a *compatible printer* must be plugged into the computer's USB port.

1. Click "Print" button and wait for `sd-devices` VM to start.
2. You will prompted to attach your printer.
3. A Print Document dialog will appear, from which you can configure different print options before printing the document.

1.13.4 Exporting submissions from the *Journalist Workstation*

Important

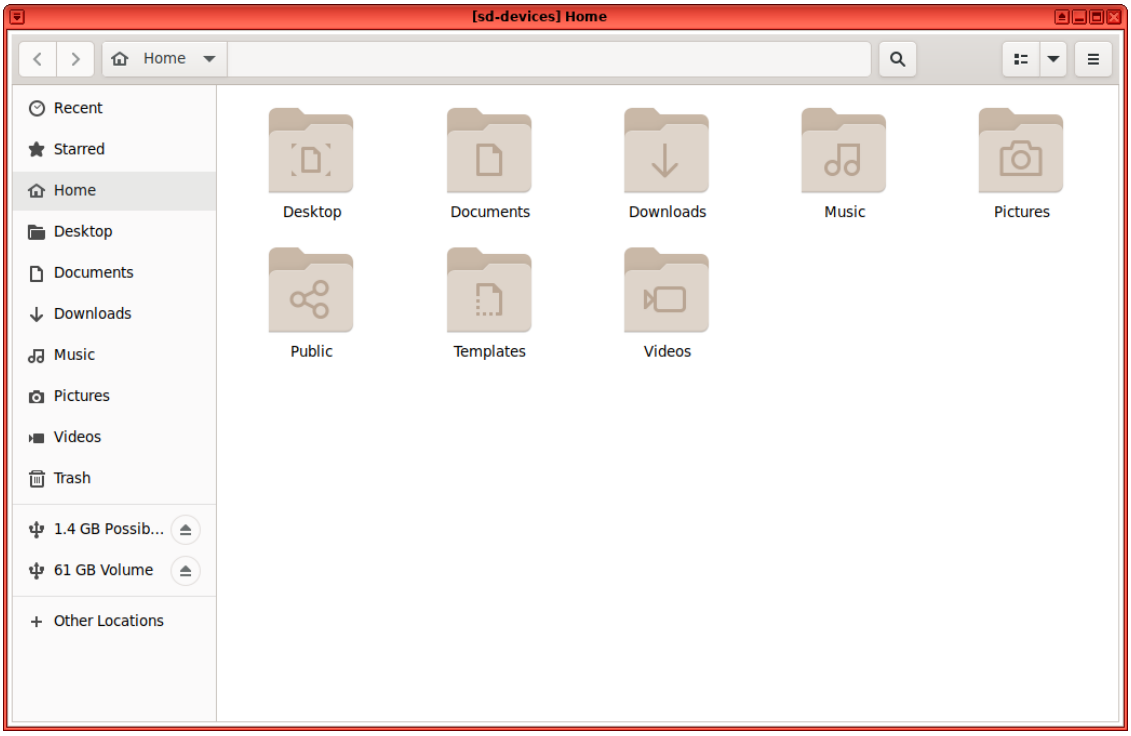
SecureDrop does not scan for or remove malware. If the file you received contains malware targeting the operating system and applications running on your everyday workstation, copying it in its original form carries the risk of spreading malware to that computer. Make sure you understand the risks, and consider other methods to export the document (e.g., print).

If you must copy a file from your *Journalist Workstation* to another computer or device in digital form, our *recommendation* is that *Journalists* are provided with an *Export Device*, drive which is encrypted using LUKS or VeraCrypt. These instructions assume that you are following the recommended workflow. If you are unsure, ask your administrator.

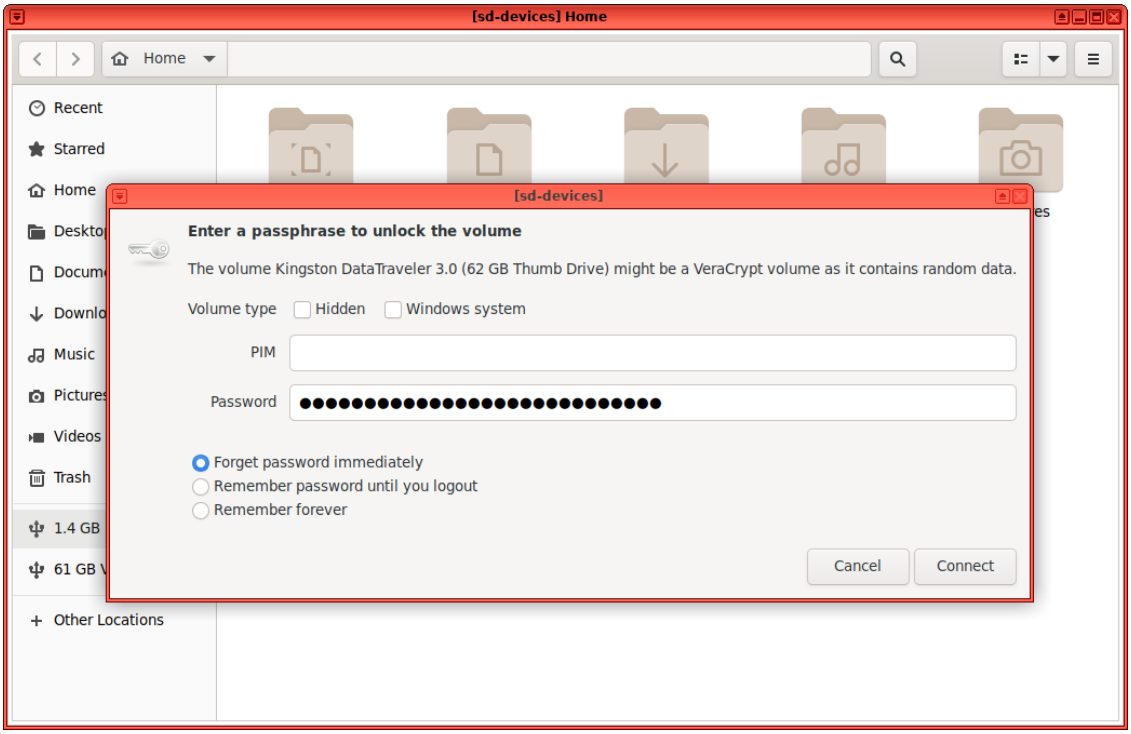
Exporting to an *Export Device*

Currently, a LUKS- or VeraCrypt-encrypted USB flash drive is required for exporting submissions.

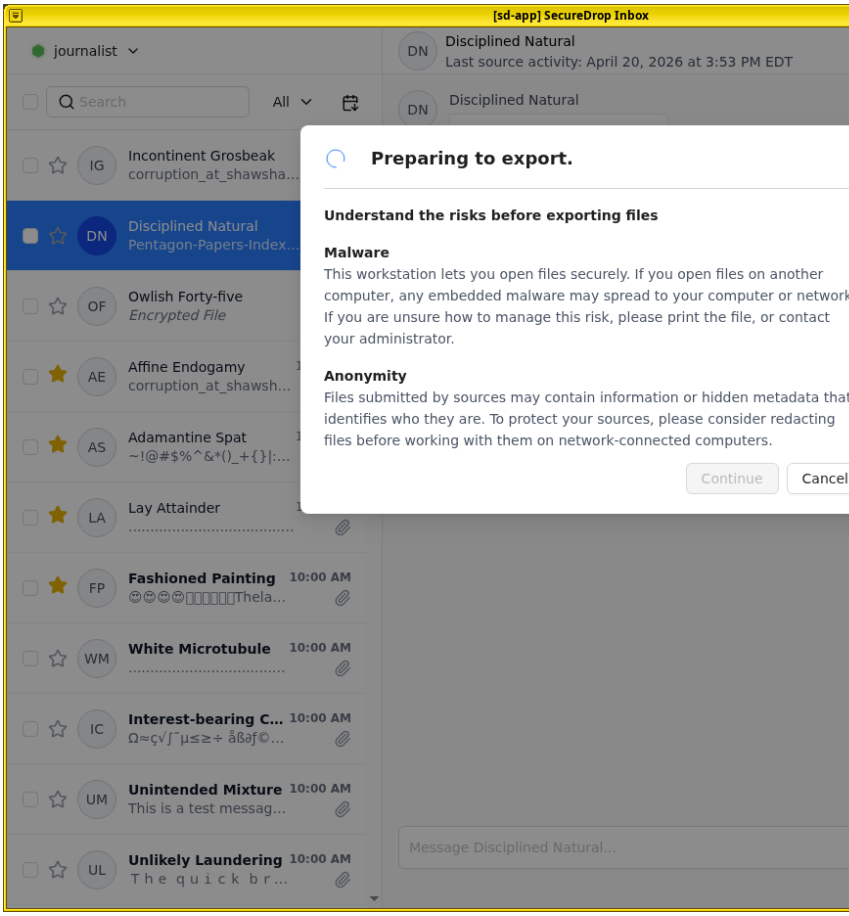
1. Insert the USB flash drive and wait for the `sd-devices` VM to start.
2. If your drive is using VeraCrypt, you will need to unlock it manually:
    1. Open the file menu by clicking on the Qubes Application menu  (in the top left), select **sd-devices** and click **Files**.
    2. In the left sidebar, there should be an entry labeled **# GB Possibly Encrypted**, click it.



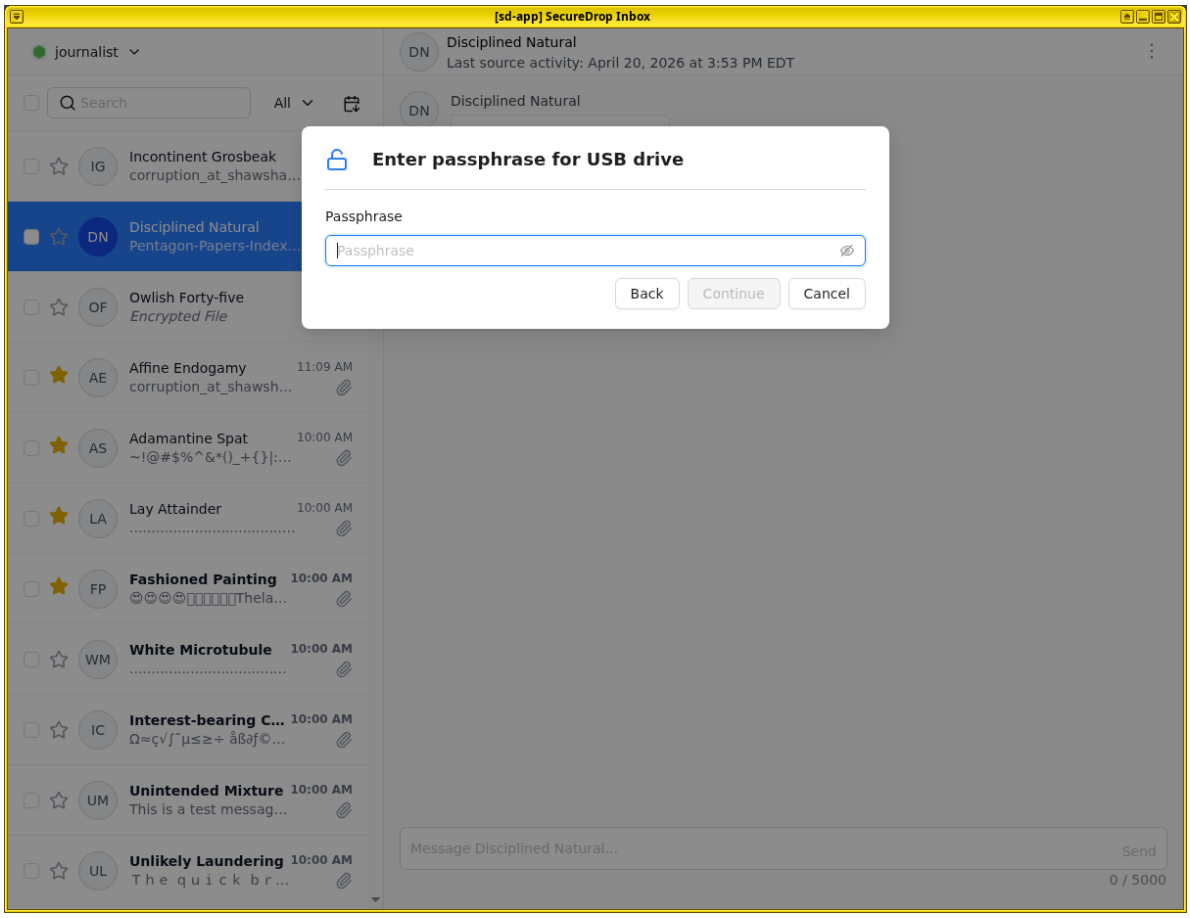
3. You will be prompted for the password configured for this USB flash drive:
   - Volume type: leave both unchecked
   - PIM: leave empty
   - Password: drive's password
   - Forget password immediately: selected



4. Click **Connect**.



3. Back in your *Source*'s conversation, click **Export**.
4. If you have not already unlocked your USB flash drive, you will be prompted for the password configured for this USB flash drive.



5. Once you see a message informing you that the export was successfully completed, you can safely unplug the USB flash drive. Alternatively, you can leave the drive plugged in and export additional files.

Decrypting and preparing to publish

Note

To decrypt a VeraCrypt drive on a Windows or Mac workstation, you need to have the *VeraCrypt* software installed. If you are unsure if you have the software installed or how to use it, ask your administrator, or see the Freedom of the Press Foundation guide for working with VeraCrypt.

To access the *Export Device* on your everyday workstation, follow these steps:

1. If your *Export Device* has a physical write protection switch, make sure it is in the *locked* position.
2. Plug the *Export Device* into your everyday workstation.
3. Launch the *VeraCrypt* application.
4. Click **Select Device** and select the *Export Device*, then click **OK**.
5. Click **Mount**.
6. Enter the passphrase for your *Export Device*. You should find this in your own personal password manager.
7. Open the *Export Device* in your operating system's file manager, and copy the contents of interest to your everyday workstation.

As a security precaution, we recommend deleting the files on the *Export Device* after each copy operation.

When you are done, switch back to the *VeraCrypt* window, and click **Dismount**.

You are now ready to write articles and blog posts, edit video and audio, and begin publishing important, high-impact work!

Tip

Check out our SecureDrop *Promotion Guide* to read about encouraging sources to use SecureDrop.

1.13.5 Safely working with submissions outside the *Journalist Workstation*

Risks from malware

SecureDrop does not scan for or remove malware in submissions you receive. There are important steps you can take to protect yourself:

1. **Keep your *Journalist Workstation* up-to-date.**
2. **Print documents from the *Journalist Workstation* instead of exporting them digitally, whenever possible.**

   Printing documents prevents the proliferation of malware to your everyday workstation, and eliminates most categories of embedded metadata. Note that printing a document may still preserve watermarks, printer codes, steganographically encoded data, or other information not visible to the naked eye.
3. **Consult with your administrator or your digital security staff before copying files digitally.**

   If you must copy a file in digital form (because of its format, the volume of information, or for other reasons), we recommend taking the time to consult with technical experts within the organization.

   Tip

   Converting files to simpler formats (e.g., PDF to PNG) can help reduce the risk of malware. Tails provides both graphical and command-line utilities that can be used for this purpose.
4. **Never scan QR codes embedded in documents using a network-connected device.**

   QR codes can contain malicious links that your device will automatically visit. This can alert third-parties to your actions, reveal the identities of your *Sources*, and breach the isolation benefits of using Qubes.

   In general, be careful when opening any links provided in a SecureDrop submission. If you are unsure if a link is safe to click, you should consult internally, or contact Freedom of the Press Foundation for assistance.
5. **Don't photograph submissions using your smartphone, and be careful with all digital photography**.

   Many smartphones are configured to back up photographs to cloud services, immediately or intermittently; newer digital cameras have similar functionality. Not all backup settings may be visible to you.

   Any digital photograph will include certain metadata by default, which may reveal sensitive information about your SecureDrop usage patterns (potentially including GPS coordinates) to anyone who gains access to the file.

Fully mitigating the risks of malware received via SecureDrop is beyond the scope of this documentation. If you have questions, you can *contact us*. Please do **NOT** disclose details about the contents of any submission you have received.

Tip

This is only a very limited introduction. Freedom of the Press Foundation publishes and maintains digital security guides for journalists, many of which relate to these topics, and offers digital security training for news organization staff.

1.14 Ending your session

When you are finished using your *Journalist Workstation*, close the SecureDrop Inbox window and shut the computer down completely. This is to take advantage of the protections of full-disk encryption, and to avoid unauthorized access to the Workstation and the files and materials on it, which include any messages and submissions that you have downloaded.

To shut down the computer, click your username in the top righthand corner of your screen, and select **Shut Down** from the menu.

1.15 Introduction for SecureDrop administrators

SecureDrop servers are managed by a systems administrator.

For larger newsrooms, there may be a team of systems admins, but at least one person within the organization will need to serve as the administrator. In some situations, such as smaller news organizations where a *Journalist* has the technical capacity to administer systems, one person can serve as both *Journalist* and administrator. When possible, we advise having a dedicated staff member serving the role of SecureDrop administrator.

The admin connects to the *Application* and *Monitor Servers* over authenticated *Onion Services*, and manages them using Ansible.

If you are considering becoming a SecureDrop administrator, below are some attributes that will be important to have:

- Experience with managing Linux-based systems from the command line.
- Proficiency with network hardware such as firewalls and switches (e.g. pfSense).
- Experience with QubesOS.
- Experience with configuration management tools such as Ansible, Salt, Chef, or Puppet.
- Ability to use and configure secure communication tools such as GPG.

We consider the first two requirements and the last three preferred attributes.

This Admin Guide covers planning, installation, deployment, and ongoing maintenance of a SecureDrop installation.

1.15.1 Responsibilities of SecureDrop administrators

The SecureDrop architecture contains multiple machines and hardened servers. While many of the installation and maintenance tasks have been automated, a skilled Linux admin is required to responsibly run the system.

As a SecureDrop administrator, it is your responsibility to:

- *install SecureDrop*
- *manage users*
- *manage the system configuration*
- *ensure that servers, firewall and workstations are kept up-to-date*
- *monitor OSSEC alerts*

- *monitor the SecureDrop team's release and security-related communications*
- apply available firmware updates to all SecureDrop hardware
- ensure that the SecureDrop environment is physically secure and monitored
- ensure that SecureDrop Workstations are kept up to date
- investigate and respond to security incidents
- schedule and perform required maintenance tasks, such as operating system upgrades
- ensure that *Journalists* adhere to the documented processes for checking SecureDrop, communicating with *Sources*, and reviewing documents
- verify the integrity of SecureDrop code
- avoid the installation of unsupported code or patches
- *decommission SecureDrop after it is no longer in use*

1.15.2 Responsibilities of the SecureDrop team

The SecureDrop team employed by Freedom of the Press Foundation (FPF) and the SecureDrop community maintain and develop the SecureDrop software, which is offered as open source software, free of charge, and at your own risk.

FPF offers *paid priority support services*. We are happy to provide assistance with installing the system, with training of administrators and *Journalists*, and with investigation of technical issues and incidents.

Note

Each SecureDrop instance is hosted and operated independently. Freedom of the Press Foundation does not offer systems administration, hosting or "remote hands" services.

When the SecureDrop team becomes aware of a security vulnerability in SecureDrop or its software dependencies, we assess the impact of the vulnerability in the context of existing security mitigations and *our threat model*. Based on this assessment, we prioritize technical work and external communications.

For high severity issues that require technical changes to SecureDrop, we will issue a point release as soon as possible. As part of issuing a release or advisory, we will post further details on the SecureDrop website and to the support portal.

In rare circumstances when a technical fix is extremely time sensitive, we may provide signed patches to impacted SecureDrop instances. Even in these cases, we ask that you never install code provided to you that is not signed using the current SecureDrop release key.

When in doubt how to resolve an issue, please avoid following technical instructions that have not been vetted by the SecureDrop team. If you encounter bugs, please report them. For sensitive matters, you can contact us via the SecureDrop Support Portal or via our contact form.

1.15.3 Managing users

Admins are responsible for managing user credentials and encouraging best practices. (See *Passphrase Best Practices*.) The admin will also have access to the *Journalist Interface*, via her own username, passphrase, and *Two-Factor Authentication* method (using a smartphone application or YubiKey).

See *User Management* for more information on adding and managing users.

1.15.4 Managing the system configuration

Admins are responsible for configuring and maintaining the system. Several tools are available to support this:

- *The Admin Interface* allows the admin to manage users and configure web interface features such as organizations logos and submission preferences
- *Server SSH access* is also available, to allow administrators to troubleshoot server issues and perform manual updates.
- *The securedrop-admin utility* is used via the *Admin VM* to configure and install SecureDrop, to perform operations including server backups and restores, and to update the server configuration after installation.

1.15.5 Keeping the system updated

The admin is responsible for ensuring that updates are applied to SecureDrop. Where possible, updates are applied automatically, but some update operations require manual intervention.

Updates: servers

The admin should be aware of all SecureDrop updates and take any required manual action if requested in the SecureDrop Release Blog (RSS feed). We also recommend registering with the SecureDrop Support Portal to stay apprised of upcoming releases.

Most often, the SecureDrop servers will automatically update via `apt`. However, occasionally you will need to take other manual steps. If you are in touch with us directly for *support*, we will let you know in advance of major releases if manual intervention will be required.

Updates: network firewall

Given all traffic first hits the network firewall as it faces the non-Tor public network, the admin should ensure that critical security patches are applied to the firewall.

Because of recent changes to the frequency and scope of security updates, we do not recommend the use of pfSense Community Edition (CE). pfSense Plus continues to receive necessary security updates on a regular basis, and is provided with the purchase of most Netgate firewalls. If you wish to use a custom firewall or alternate option, we recommend using an OPNSense-based solution.

If you're using one of the network firewalls recommended by FPF, you can subscribe to email updates from the Netgate homepage or follow the Netgate blog to be alerted when releases occur. If critical security updates need to be applied, you can do so through the firewall's pfSense WebGUI.

Refer to our *Keeping pfSense up to date* documentation or the official pfSense Upgrade Docs for further details on how to update the suggested firewall.

No matter which vendor you go with, you should make it a priority to stay informed of potential updates to your network firewall.

Updates: workstations

SecureDrop Workstation includes an updater application that runs automatically on startup, checks for Qubes and SecureDrop updates, and prompts the user to apply them if found. Given the sensitive nature of the system, it is critical that updates are applied when available. Administrators should ensure that users are aware of this requirement, and should periodically check to ensure that the system is up to date.

1.15.6 Monitoring OSSEC alerts

SecureDrop uses OSSEC to monitor the servers for unusual activity caused by system configuration issues or security breaches. The admin should decrypt and read all OSSEC alerts. Report any suspicious events to FPF through the SecureDrop Support Portal. See the *OSSEC Guide* for more information on common OSSEC alerts.

Warning

Do not post logs or alerts to public forums without first carefully examining and redacting any sensitive information.

1.15.7 Monitoring SecureDrop-related communications

Release announcements and security advisories are posted to the SecureDrop blog, which is also available as an RSS feed. You can also follow us on our social media accounts (Twitter and Mastodon).

We strongly recommend *joining the SecureDrop support portal*. As a member of the support portal, you will receive email notifications related to all major announcements, and you can open tickets in case of technical issues. Membership is free of charge.

1.15.8 Installation support

Any organization can install SecureDrop for free and also make modifications because the project is open source.

Because the installation and operation are complex, and because SecureDrop can only be as secure as the operational security practices followed by its users, Freedom of the Press Foundation will also help organizations install SecureDrop and train *Journalists* and administrators.

If you would like to work with Freedom of the Press Foundation on your SecureDrop installation, please reach out to us. We do ask news organizations that can afford to pay for installation support, training and maintenance to do so.

As part of priority support agreements and on a pro-bono basis for smaller news organizations, Freedom of the Press Foundation will visit your offices, help set up SecureDrop and train *Journalists* to use it. (For pro-bono support, we request that our travel costs are covered.)

Note

SecureDrop wants your feedback! Confused by something in our documentation? Let us know by opening an issue on GitHub or in our Gitter channel.

1.16 Installation overview

1.16.1 Migrating from a Tails-based SecureDrop

If you are migrating from an older SecureDrop, using the separate Tails-based *Secure Viewing Station*, *Journalist workstation* and *Admin Workstation* USB flash drives, then skip to the *Migration Overview*.

1.16.2 Setting expectations

SecureDrop is a technical tool. It is designed to protect *Journalists* and *Sources*, but no tool can guarantee safety. This guide will instruct you in installing and configuring SecureDrop, but it does not explain how to use it safely and effectively. Put another way: at the end of this guide, you will have built a car; you will not know how to drive. The *Deployment Guide* contains best practices for working with SecureDrop. Make sure to read it after completing the installation.

Setting up SecureDrop is a multi-step process, where each step builds on the steps that come before it. It's important that you treat the installation as a complete process, making sure not to skip any portions of the install guide or jump ahead to later content.

Once you have all the necessary hardware, *setting up SecureDrop* will take at least a day's work.

We recommend that you set aside at least a week to *complete and test* your setup.

1.16.3 Tracking your progress

To assist in the installation process, we offer a SecureDrop Installation Worksheet, which you can print out and complete as you go. Only complete this worksheet on paper, never electronically.

It is **critical** that you destroy this worksheet when your installation is complete and all of your passphrases have been safely stored in a password manager.

Warning

Remember to destroy the SecureDrop Installation Worksheet after the installation is complete.

1.16.4 Technical summary

This installation guide will walk you through the process of setting up the computers and services needed for a functional SecureDrop.

During this process, you'll set up at least four devices:

- ***Admin Workstation*:**
  A laptop running the QubesOS operating system configured as an *Admin Workstation*, that you use to install and administer SecureDrop on the servers via SSH. If necessary (i.e. in a small newsroom), the same *SecureDrop Workstation* used for administration may be used as a *Journalist Workstation* by *Journalists* to decrypt, view, and export submitted documents. For a larger newsroom, you may set up additional *Journalist Workstations* as needed for *Journalist* use.
- ***Application Server*:**
  An Ubuntu server running two segmented Tor hidden services. The *Source* connects to the *Source Interface*, a public-facing Tor *Onion Service*, to send messages and documents to the *Journalist*. The *Journalist* connects to the *Journalist Interface*, an authenticated Tor *Onion Service*, using SecureDrop Inbox on a *Journalist Workstation* to download encrypted documents and respond to *Sources*.
- ***Monitor Server*:**
  An Ubuntu server that monitors the *Application Server* with OSSEC and sends email alerts.
- **Network Firewall**
  A hardware firewall dedicated to your SecureDrop installation.

A summary of the major steps is as follow:

1. Acquire compatible hardware.
2. Prepare email accounts and GPG keys for alert emails.
3. Prepare an *Admin Workstation* laptop.
4. Set up the KeePassXC password manager on the *Admin Workstation*.
5. Install and configure the dedicated network firewall from the *Admin Workstation*.
6. Prepare the (*Application* and *Monitor*) servers.
7. Install SecureDrop on the servers from the *Admin Workstation*.

8. Complete local configuration of the *Admin Workstation*.
9. Create the first Admin user.
10. Test the installation.

Optionally: #. Prepare additional *Journalist Workstations* for use by *Journalists*. #. Prepare encrypted *Export Devices*.

1.16.5 Minimum security requirements for a *SecureDrop Workstation*

A *SecureDrop Workstation* (either an *Admin Workstation* or a *Journalist Workstation*) contains both a copy of the *Submission Private Key*, and encrypted and decrypted messages and submissions. It's critical to ensure that appropriate security practices are applied to a *SecureDrop Workstation*.

- *SecureDrop Workstations* should be stored in a secure and locked room, with access restricted to users and administrators. The room may be monitored externally, but there should be no internal monitoring.
- A wired Internet connection that does not restrict Tor must be available for the *SecureDrop Workstation*. This connection should either be dedicated to *SecureDrop Workstation*, or should be on a fully segregated subnet from the rest of the corporate network.
- Users should not bring other electronic devices into the room, with the exception of smartphones used for 2FA token generation. While in the room, smartphones should be set to airplane mode, and should not be used for any purpose other than 2FA.

1.16.6 Minimum security requirements for the SecureDrop servers

- The *Application* and *Monitor Servers* should be dedicated physical machines, not virtual machines.
- A trusted location to host the servers. The servers should be hosted in a location that is owned or occupied by the organization to ensure that their legal department can not be bypassed with gag orders.
- The SecureDrop servers should be on a separate internet connection or completely segmented from the corporate network, such as a dedicated subnet with DENY rules for all traffic to and from the corporate LAN.
- All traffic from the corporate network should be blocked at the SecureDrop's point of demarcation.
- Video monitoring should be recorded of the server area and the organizations safe.
- An established monitoring plan and incident response plan. Who will receive the OSSEC alerts and what will their response plan be? These should cover technical outages and a compromised environment plan.

1.17 Hardware

This document outlines the required hardware components necessary to successfully install and operate a SecureDrop instance, and recommends some specific components that we have found to work well. If you have any questions, please *contact the SecureDrop Support team*.

1.17.1 Hardware overview

For an installation of SecureDrop, you must acquire:

- 2 computers (with storage drives) to use as the SecureDrop servers.
- A mouse, keyboard, and monitor (along with any necessary dongles or adapters) for installing the servers.
- At least 1 dedicated physical laptop for the *SecureDrop Workstation*.
- A dedicated network firewall with at least 4 NICs.
- At least 3 ethernet cables.

- At least 1 USB flash drive for OS installation media, and at least 1 more USB flash drive if needed as an *Export Device*.

Additionally, you may want to consider the following purchases:

- a printer without wireless network support, to use in combination with the *SecureDrop Workstation*.
- an external hard drive for server backups.
- a USB flash drive to store backups of your *SecureDrop Workstation*.
- a security key for HOTP authentication, such as a YubiKey, if you want to use hardware-based *Two-Factor Authentication* instead of a mobile app.
- a USB flash drive with a physical write protection switch, or a USB write blocker, if you want to mitigate the risk of introducing malware from your network to your *SecureDrop Workstation* during repeated use of an *Export Device*.

Tip

While a printer is not required, we highly recommend it. Printing documents is generally far safer than copying them in digital form. See our guide to working with documents for more information.

1.17.2 Advice for users on a tight budget

If you cannot afford to purchase new hardware for your SecureDrop instance, we encourage you to consider re-purposing existing hardware to use with SecureDrop. If you are comfortable working with hardware, this is a great way to set up a SecureDrop instance for cheap.

Since SecureDrop's throughput is significantly limited by the use of Tor for all connections, there is no need to use top of the line hardware for any of the servers or the firewall.

We recommend against re-purposing Apple Macintosh laptops and desktops, due to incompatibility with Qubes OS.

If you choose to use recycled hardware, you should of course consider whether or not it is trustworthy; making that determination is outside the scope of this document.

1.17.3 Required hardware

Servers

- *Application Server*: 1 physical server to run the SecureDrop web services.
- *Monitor Server*: 1 physical server which monitors activity on the *Application Server* and sends email notifications to an admin.

We recommend using NUCs for the servers and routinely test new models for compatibility. NUCs ("Next Unit of Computing") are comparatively inexpensive, compact, quiet, and low-power devices, which makes them suitable for deployment in a wide range of environments. Originally produced by Intel, ASUS has taken over production beginning with the 14th generation.

There are a variety of models to choose from. We currently recommend the 11th through 14th generation NUC models listed below.

Note

If using non-recommended hardware, ensure you remove as much extraneous hardware as physically possible from your servers. This could include: speakers, cameras, microphones, fingerprint readers, wireless, and Bluetooth cards.

NUCs typically come as kits, and some assembly is required. You will need to purchase the RAM and hard drive separately for each NUC and insert both into the NUC before it can be used. We recommend:

- 2x 240GB SSDs (2.5" or M.2, depending on your choice of kit)
- 1x memory kit of compatible 2x8GB sticks - You can put one 8GB memory stick in each of the servers.

We are often asked if it is acceptable to run SecureDrop on cloud servers (e.g. Amazon EC2, DigitalOcean, etc.) or on dedicated servers in third-party datacenters instead of on dedicated hardware hosted in the organization. This request is generally motivated by a desire for cost savings and/or convenience. However: we consider it **critical** to have dedicated physical machines hosted within the organization for both technical and legal reasons:

- While the documents are stored encrypted at rest (via PGP) on the SecureDrop *Application Server*, the documents hit server memory unencrypted (unless the source used the GPG key provided to encrypt the documents first before submitting), and are then encrypted in server memory before being written to disk. If the machines are compromised then the security of source material uploaded from that point on cannot be assured. The machines are hardened to prevent compromise for this reason. However, if an attacker has physical access to the servers either because the dedicated servers are located in a datacenter or because the servers are not dedicated and may have another virtual machine co-located on the same server, then the attacker may be able to compromise the machines. In addition, cloud servers are trivially accessible and manipulable by the provider that operates them. In the context of SecureDrop, this means that the provider could access extremely sensitive information, such as the plaintext of submissions or the encryption keys used to identify and access the *Onion Services*.
- In addition, attackers with legal authority such as law enforcement agencies may (depending on the jurisdiction) be able to compel physical access, potentially with a gag order attached, meaning that the third party hosting your servers or VMs may be legally unable to tell you that law enforcement has been given access to your SecureDrop servers.

One of the core goals of SecureDrop is to avoid the potential compromise of sources through the compromise of third-party communications providers. Therefore, we consider the use of virtualization for production instances of SecureDrop to be an unacceptable compromise and do not support it. Instead, dedicated servers should be hosted in a physically secure location in the organization itself. While it is technically possible to modify SecureDrop's automated installation process to work on virtualized servers (for example, we do so to support our CI pipeline), doing so in order to run it on cloud servers is at your own risk and without our support or consent.

14th-gen NUC

We have tested and can recommend the ASUS NUC14RVH. It provides both 22x80 and 22x42 M.2 ports for NVMe SSD storage, as well as a 2.5 inch drive bay for a SATA hard drive or SSD (if using this slot, we recommend choosing an SSD).

The NUC14's AX211 wireless hardware is not removable. Before installation of the RAM and storage, we recommend that you disconnect the wireless antennae leads from the AX211 component. They're the wires highlighted in the red box in the picture. Cover the free ends with electrical tape after disconnecting them.

Note

The wireless card is located underneath the NVMe port

13th-gen NUC

We have tested and can recommend the ASUS NUC13ANHi5. It provides two M.2 SSD storage options: a 22x80 port for an NVMe drive, and a 22x42 port for a SATA drive. It also has a 2.5 inch drive bay for a SATA hard drive or SSD (if using this slot, we recommend choosing an SSD).

The NUC13's AX211 wireless hardware is removable. Doing so requires the use of a 5mm nut driver. Before installation of the RAM and storage, we recommend that you remove the wireless card and disconnect the wireless antennae

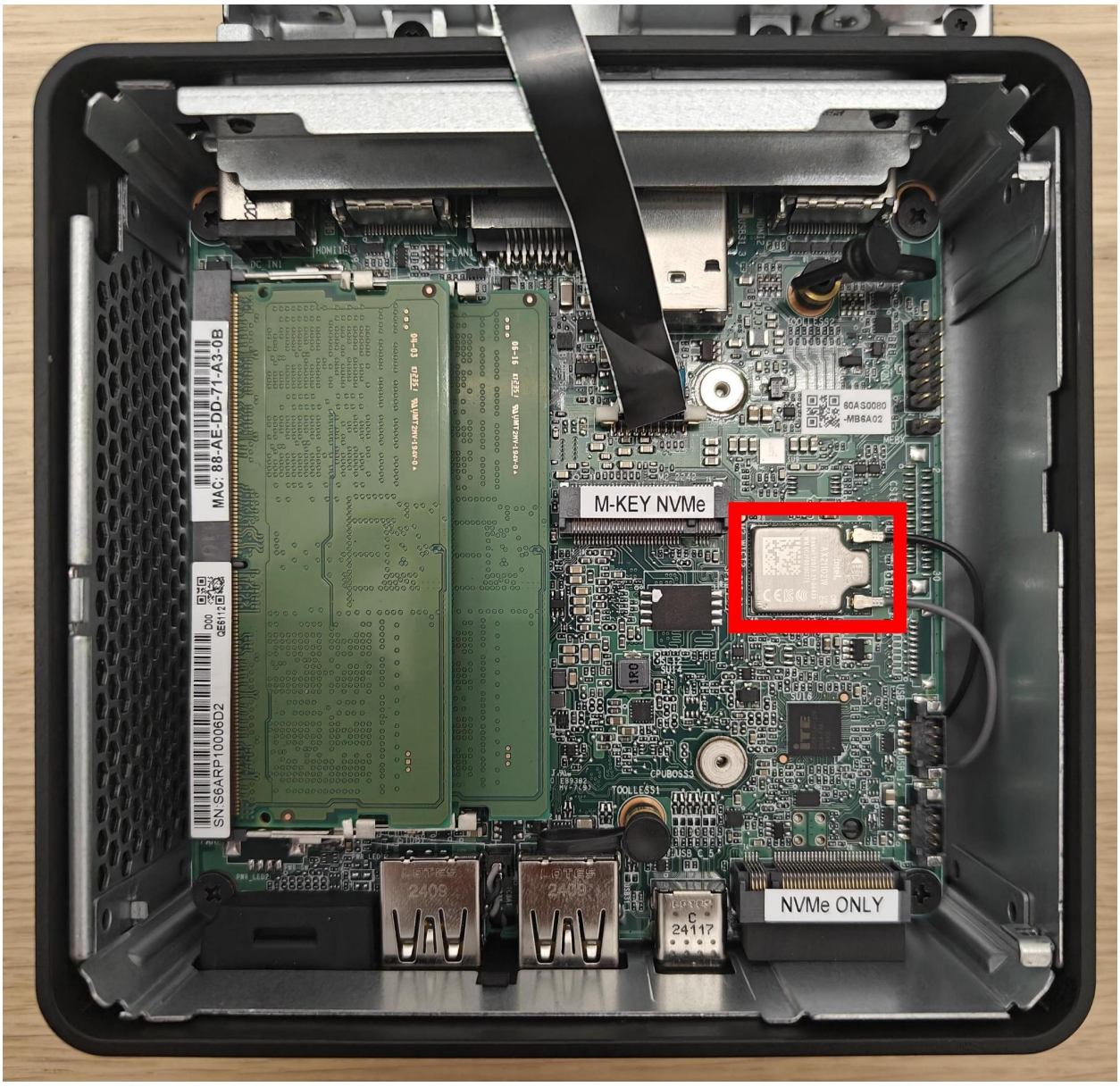


Fig. 1: The location of the wireless card within the NUC14

leads from the AX211 component. Be sure to cover the free ends with electrical tape after disconnecting them.

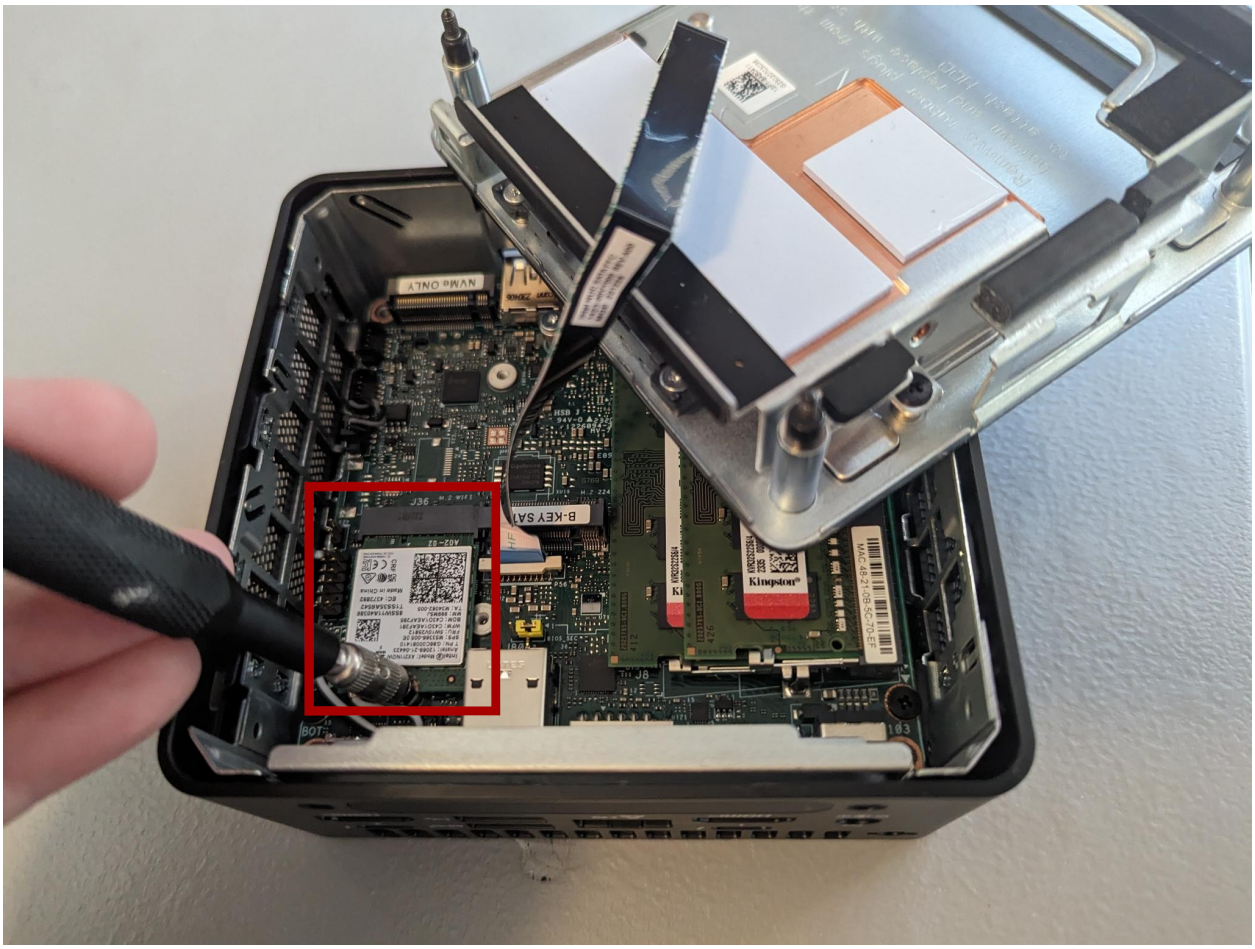


Fig. 2: The location of the wireless card within the NUC13

Note

The wireless card is located underneath the 22x80 NVMe port

12th-gen NUC

We have tested and can recommend the NUC12WSKi5. It provides two M.2 SSD storage options: a 22x80 port for an NVMe drive, and a 22x42 port for a SATA drive.

The NUC12's AX211 wireless hardware is removable. Doing so requires the use of a 5mm nut driver. Before installation of the RAM and storage, we recommend that you remove the wireless card and disconnect the wireless antennae leads from the AX211 component. Be sure to cover the free ends with electrical tape after disconnecting them.

11th-gen NUC

We have tested and can recommend the Intel NUC11PAHi3. It provides two storage options: M.2 SSD storage and a 2.5" secondary storage option (SSD or HDD).

The NUC11's AX201 wireless hardware is not removable. Before installation of the RAM and storage, we recommend

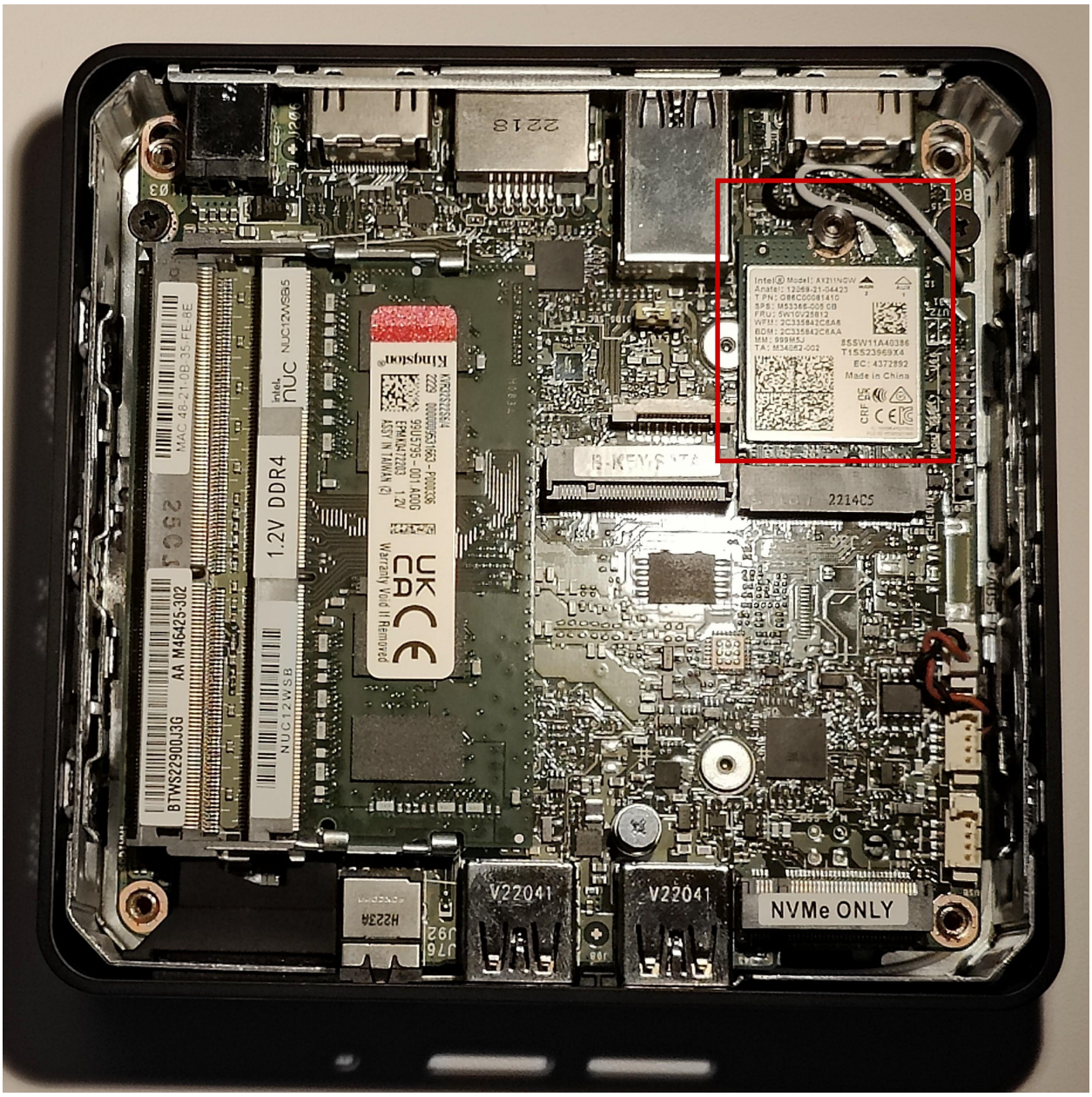
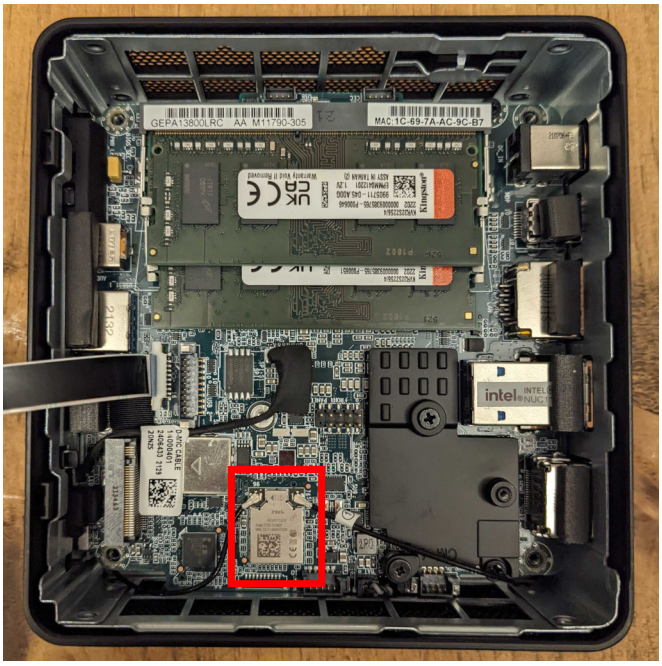


Fig. 3: The location of the wireless card within the NUC12

that you disconnect the wireless antennae leads from the AX201 component. They're the black wires highlighted in the red box in the picture. Cover the free ends with electrical tape after disconnecting them.



Before the initial OS installation, boot into the BIOS by pressing **F2** at startup and adjust the system configuration:

Workstations

In order to install and use *SecureDrop Workstation*, you will need a Qubes-compatible computer with the following specifications:

- 64-bit Intel processor with virtualization support
- a minimum of 32GB RAM
- sufficient disk space for the Qubes OS base install and SecureDrop Workstation VMs (a 128GB or greater SSD is recommended)

We recommend against a device that requires an external USB keyboard or other externally-connected devices, for security reasons. In practice this usually means that you should run SecureDrop Workstation on a Qubes-compatible laptop. Not all laptops support Qubes, and some may require additional customization. We recommend (in order) either a Qubes-certified laptop, one of the laptop models we use for development and testing, or a computer from the community-maintained Qubes Hardware compatibility list.

Qubes-certified laptops

Qubes-certified laptops are certified and tested against Qubes major releases. They must support additional security features beyond the minimal requirements above, such as the use of coreboot in place of proprietary firmware. Where possible, we recommend that you use a Qubes-certified laptop with `coreboot` for SecureDrop Workstation. A full list of certified computers can be found on the Qubes OS Certified Hardware page.

> **Note**
>
> Some certified computers also support the use of Heads with `coreboot`, for additional protection against advanced attacks during the boot process. Heads adds a layer of complexity to the overall user

experience, but may make sense for you as an option if you have an expectation of those kinds of threats. If you have questions about Heads, or other hardware choices, contact us via Signal`_.

FPF-tested laptops

In addition to Qubes-certified devices, we develop and test using Qubes-compatible laptops from other vendors. The following models may be used for SecureDrop Workstation, though some level of additional configuration may be required.

Framework 13 (Intel Core Ultra Series 1)

The Framework 13 laptop with an Intel Core Ultra Series 1 processor is a recommended option for the SecureDrop Workstation beginning with Qubes 4.2.

You can either order a preassmbled system, or you can customize your build and assemble the laptop yourself once it is delivered, which is useful as either a cost-saving measure or in the event that you wish to customize the ports or internal components.

Framework laptops are designed to be repairable, customizable, and user-servicable, and have grown to be a popular choice with Qubes users and SecureDrop developers.

You will want to ensure you are using the latest BIOS version available. Instructions for checking the BIOS version and performing an upgrade for the Intel Core Ultra Series 1 models can be found on this page in the Framework knowledgebase.

Note

You'll want to be sure to install Qubes OS using the kernel-latest option, available from the initial boot menu (GRUB) prior to booting to the Qubes OS installer.

Framework 13 (13th-generation)

The Framework 13 laptop with a 13th generation Intel processor is a recommended option for the SecureDrop Workstation beginning with Qubes 4.2.

You can either order a preassmbled system, or you can customize your build and assemble the laptop yourself once it is delivered, which is useful as either a cost-saving measure or in the event that you wish to customize the ports or internal components.

Framework laptops are designed to be repairable, customizable, and user-servicable, and have grown to be a popular choice with Qubes users and SecureDrop developers.

You will want to ensure you are using the latest BIOS version available. Instructions for checking the BIOS version and performing an upgrade for the 13th generation models can be found here in the Framework knowledgebase.

Lenovo ThinkPad X1 Carbon (10th-generation)

The 10th-generation ThinkPad X1 Carbon **with a 12th-generation Intel Core processor** is a recommended option for the SecureDrop Workstation beginning with Qubes 4.1. If you plan to use it, you will want to ensure the BIOS is up-to date by following these instructions: *Automatic BIOS updates*.

You'll need to have a USB-to-Ethernet adapter on hand in order to *apply Qubes updates*, which will enable Wi-Fi and fix glitchy video rendering and cursor performance.

Lenovo ThinkPad T14 (2nd-generation)

The 2nd-generation ThinkPad T14 **with an 11th-generation Intel Core processor** is a recommended option for the SecureDrop Workstation beginning with Qubes 4.1. If you plan to use it, you will want to ensure the BIOS is up-to date by following these instructions: *Automatic BIOS updates*.

The Ethernet and Wi-Fi controllers may not work without one-time manual configuration, as documented here.

The Qubes Hardware Compatibility List (HCL)

The Qubes Hardware Compatibility List (HCL) is a community-maintained list of hardware that has been tested by Qubes users. It consists of individual reports generated and submitted by Qubes users across the world. Anyone can attempt to install Qubes on their computer, then report back on whether or not it can be installed, if there are any issues, and overall, what the experience is like.

There are some benefits to this list:

- A much wider selection of hardware is tested, because anyone can contribute to the list
- There are sometimes multiple reports for a particular system, which lets you compare and feel confident the results are consistent
- It tells you exactly what is and isn't working within the system, so you can decide if a device you own will function well enough to suit your needs
- Devices get tested across many different configurations and Qubes versions

However, there are some things to consider:

- Reports are not verified for their accuracy by either the Qubes team or Freedom of the Press Foundation
- Reports correspond to a specific Qubes OS version, and may not reflect breaking changes or expanded hardware support in the most recent Qubes OS version
- It's important that you update the BIOS of your laptop prior to installing SecureDrop Workstation: for more details see *Automatic BIOS updates*

For the best experience, we recommend choosing a Qubes-certified laptop, or a laptop that we have directly tested (in that order); however, if none of those suit your needs, or if you want to see if your existing hardware might be Qubes compatible, the HCL is a good choice.

Network firewall

You will need one physical computer that is used as a dedicated firewall for the SecureDrop servers.

We recommend a 4 NIC network firewall and currently provide setup instructions for pfSense and OPNSense. Suitable models include:

- the Protectli Vault 4-Port, running OPNSense configured with coreboot.
- the Netgate SG-4100 running pfSense Plus.
- the Netgate SG-6100 running pfSense Plus. This device is overspecced for SecureDrop's purposes, but can be used if the other cheaper firewalls can't be procured.

An acceptable alternative that requires more technical expertise is to *configure an existing hardware firewall*.

Two-factor device

Two-Factor Authentication is used when connecting to different parts of the SecureDrop system. Each admin and each *Journalist* needs a two-factor device. We currently support two options for *Two-Factor Authentication*:

- Your existing smartphone with an app that computes TOTP codes (e.g. FreeOTP for Android and for iOS).

- A dedicated hardware dongle that computes HOTP codes (e.g. a YubiKey).

Tip

We recommend using FreeOTP (available for Android and for iOS) to generate two-factor codes because it is Free Software. However, if it does not work for you for any reason, alternatives exist:

- Google Authenticator for Android and iOS (proprietary)
- authenticator for the desktop (Free Software)

USB flash drives

Journalists need physical media (known as the *Export Device*) to copy submissions to their everyday workstation.

Our standard recommendation is to use USB flash drives, in combination with volume-level encryption and careful data hygiene. We also urge the use of a secure printer or similar analog conversions to export documents from the *SecureDrop Workstation*, whenever possible.

You may want to consider enforcing write protection on USB flash drives when only read access is needed. We encourage you to evaluate these options in the context of your own threat model. When it is consistently applied and correctly implemented in hardware, write protection can prevent the spread of malware from the computers used to read files stored on an *Export Device*. The two main options to achieve write protection of USB flash drives are:

- drives with a built-in physical write protection switch
- a separate USB write blocker device as used in forensic applications.

For USB flash drives with physical write protection, we have tested the Kanguru SS3, and it works well with and without encryption.

It is especially advisable to enable write protection before attaching an *Export Device* to an everyday workstation that lacks the security protections of the Tails operating system.

Please review our *setup guide* for additional background on setting up *Export Devices*.

We also recommend buying an additional USB flash drive for making regular backups of your *SecureDrop Workstations*.

One thing to consider is that you are going to have *a lot* of USB flash drives to keep track of, so you should consider how you will label or identify them and buy drives accordingly. Drives that are physically larger are often easier to label (e.g. with tape, printed sticker or a label from a labelmaker).

Monitor, keyboard, mouse

You will need these to do the initial installation of Ubuntu on the *Application* and *Monitor Servers*.

1.17.4 Optional hardware

This hardware is not *required* to run a SecureDrop instance, but most of it is still recommended.

Printers

There are several requirements for a printer to be compatible with SecureDrop Workstation. Your printer should:

1. Support **driverless printing** standards
2. Have a **USB port**
3. Be offline, or at least have **no WiFi**

These requirements are expanded below.

Driverless

SecureDrop Workstation implements driverless IPP printing to support a large selection of modern printers. Compatible printers can be easily identified by their support for the Apple AirPrint or Moipra standards:

You may consult Apple's list of printers that support AirPrint, Moipra's list of certified products, or OpenPrinting's list of printers supporting driverless printing.

USB ports

SecureDrop Workstation only supports printing over USB, so ensure the printer you select has a **USB port**.

Note

In rare cases, an AirPrint or Moipra-compatible printer with a USB port may not actually support IPP-over-USB, which is required for SecureDrop to use the printer. Check with the manufacturer if in doubt.

Offline

To maintain the isolation of SecureDrop Workstation, it is essential that your printer not be shared with other computers and networks.

- Select a compatible printer with **no WiFi**. A printer that connects with USB only is best if you can find one, but compatible USB printers lacking *both* Ethernet and WiFi are rare.
- In the case of a printer with Ethernet and/or WiFi, **keep the printer offline** and **disabling WiFi** (if present).
- Use this printer exclusively with SecureDrop Workstation and do not connect it directly to other computers.

Backup storage

It's useful to run periodic backups of the servers in case of failure. We recommend buying an external hard drive to store server backups.

Important

Like all storage media associated with SecureDrop, this drive should be encrypted and protected with a secure passphrase. We recommend using the tools built into Tails to encrypt the drive using LUKS.

If you are planning to use hardware RAID and/or hardware-based encryption, we recommend that you research Tails compatibility before a procurement decision.

1.17.5 Hardware end-of-life

No matter what hardware you decide to use, it's important to be mindful of how long it will continue to receive security updates. Given the security requirements for a SecureDrop instance, any hardware that is no longer receiving security updates from the manufacturer will become more and more vulnerable over time. Once your hardware has reached its end-of-life (EOL), we recommend upgrading to newer, supported hardware.

For the server, we previously recommended the NUC10i5FNH, NUC8i5BEK, and NUC7i5BNH. If you are still using one of these models, we recommend replacing them with one of the newer NUC models listed above.

For the hardware we recommend, you can find a list of end-of-life dates below:

Hardware	End-of-Life (EOL)
ASUS NUC14RVH	Not yet confirmed
ASUS NUC13ANHi5	Not yet confirmed
Intel NUC12WSKi5	April 05, 2026
Intel NUC11PAHi3	September 30, 2026
Thinkpad T Series	EOL dates vary; consult with manufacturer
TekLager APU4D4	Not yet confirmed
Netgate SG-4100	Not yet confirmed (will be 2 years after sales stop)
Netgate SG-6100	Not yet confirmed (will be 2 years after sales stop)

1.18 Passphrases overview

Each individual with a role (admin or *Journalist*) at a given SecureDrop instance must generate and retain a number of strong, unique passphrases. The section is an overview of the passphrases, keys, two-factor secrets, and other credentials that are required for each role in a SecureDrop installation.

Ideally, each admin and *Journalist* would only have to remember the passphrases to unlock the encrypted storage on their *Journalist Workstation* laptop.

1.18.1 Administrator

The administrator will be using an *Admin Workstation* configured to connect to the *Application Server* and the *Monitor Server* using Tor and SSH. The tasks performed by the admin will require the following set of credentials and passphrases:

- The Qubes full disk encryption (FDE) password of the *Admin Workstation*, required to unlock system storage on boot.
- The Qubes system user password for the *Admin Workstation*, required to log in.
- Additional credentials, which we recommend adding to Tails' KeePassXC password manager during the installation:
  - The *Application Server* and *Monitor Server* admin username and password (required to be the same for both servers).

- The network firewall username and password.
- The SSH private key and, if set, the key's passphrase.
- The *OSSEC Alert Public Key*.
- The admin's personal GPG public key, if you want to potentially encrypt sensitive files to it for further analysis.
- The account details for the destination email address for OSSEC alerts.
- The *Onion Services* values required to connect to the *Application* and *Monitor Servers*.

The admin will also need to have a way to generate *Two-Factor Authentication* codes.

Tip

We recommend using FreeOTP (available for Android and for iOS) to generate two-factor codes because it is Free Software. However, if it does not work for you for any reason, alternatives exist:

- Google Authenticator for Android and iOS (proprietary)
- authenticator for the desktop (Free Software)

And the admin will also have the following two credentials:

- The secret code for the *Application Server*'s *Two-Factor Authentication*.
- The secret code for the *Monitor Server*'s *Two-Factor Authentication*.

1.18.2 *Journalist*

The *Journalist* will be using a *Journalist Workstation* to view submissions with SecureDrop Inbox. The tasks performed by the *Journalist* will require the following set of passphrases:

- The Qubes full disk encryption (FDE) password of the *Journalist Workstation* they use, required to unlock system storage on boot.
- The Qubes system user password for the *Journalist Workstation* they use, required to log in.

The *Journalist* will also need to have a two-factor authenticator, such as an Android or iOS device with FreeOTP installed, or a YubiKey. This means the *Journalist* will also have the following credential:

- The secret code for the *Journalist*'s *Two-Factor Authentication*.

Export Device

We recommend using encrypted USB flash drives for transferring files off of the *Journalist Workstation*.

For every export operation, the user will need to enter the USB flash drive's encryption passphrase at least twice (on the computer they're copying from, and on the computer they're copying to). To make it easy for them to find the passphrase, we recommend storing it in the *Journalist*'s own existing password manager, which should be accessible using their smartphone.

If your organization is not using a password manager already, please see the Freedom of the Press Foundation guide to choosing one.

1.18.3 Passphrase best practices

All SecureDrop users—*Sources*, *Journalists*, and admins—are required to memorize at least one passphrase. This section describes best practices for passphrase management in the context of SecureDrop.

1. **Do** memorize your passphrase.
2. If necessary, **do** write your passphrase down temporarily while you memorize it.

   Caution

   Do store your written passphrase in a safe place, such as a safe at home or on a piece of paper in your wallet. **Do** destroy the paper as soon as you feel comfortable that you have the passphrase memorized. **Do not** store your passphrase on any digital device, such as your computer or mobile phone.

3. **Do** review your passphrase regularly. It's easy to forget a long or complex passphrase if you only use it infrequently.

   Tip

   We recommend reviewing your passphrase (e.g. by ensuring that you can log in to your SecureDrop account) on at least a monthly basis.

4. **Do not** use your passphrase anywhere else.

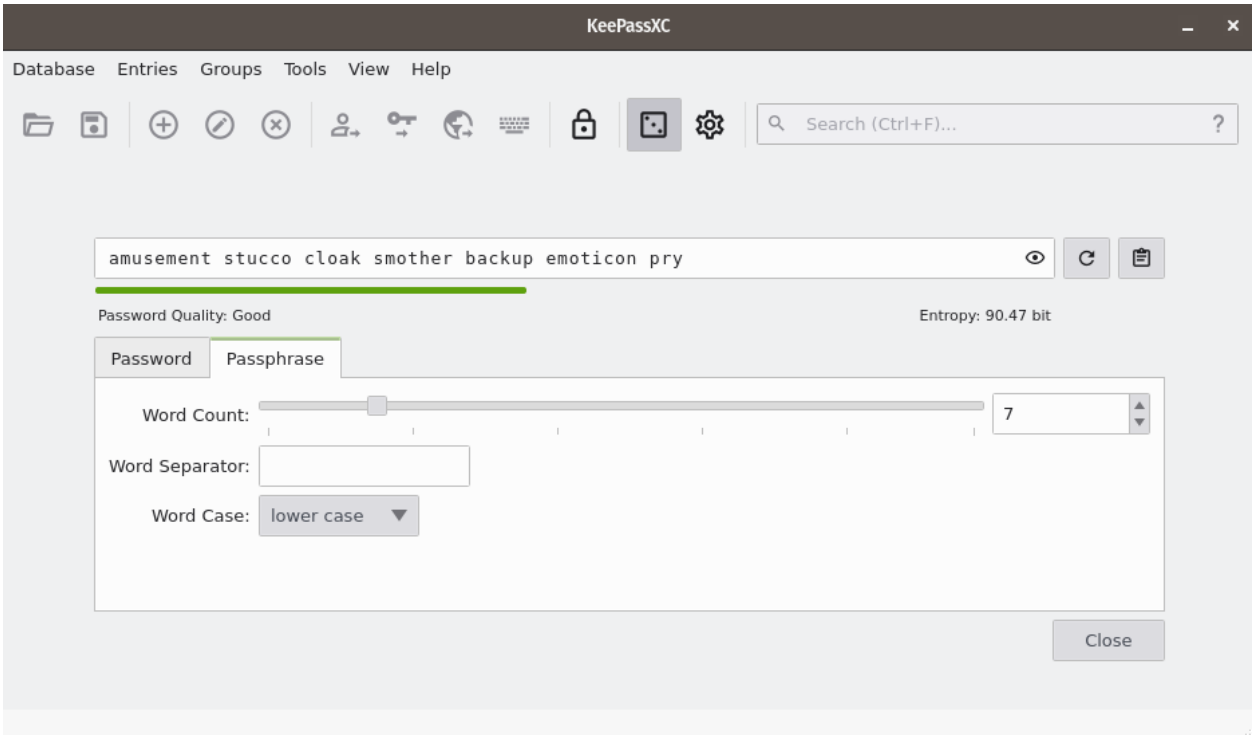
   If you use your SecureDrop passphrase on another system, a compromise of that system could theoretically be used to compromise SecureDrop. You should avoid reusing passphrases in general, but it is especially important to avoid doing so in the context of SecureDrop.

How to generate a strong, unique passphrase

We recommend using a unique, 7-word passphrase for each case described above. We encourage each end user to use KeePassXC, an easy-to-use password manager included in QubesOS, to generate and retain strong and unique passphrases. The SecureDrop installation includes a template that you can use to initialize this database, which will be explained when you set up your first **Admin Workstation**.

Using KeePassXC to generate a passphrase

To create a random passphrase using KeePassXC, launch the application, then click the **dice icon**. Then click the **Passphrase** tab and set the **Word Count** to 7. You can optionally set a **Word Separator**, for example a space or hyphen.



1.19 Prepare email accounts

SecureDrop sends different alerts by PGP-encrypted email. Before installing SecureDrop, you must select or prepare the email accounts where you would like these alerts to be sent. In the case of OSSEC alerts (which you must set up), configuring an SMTP relay is also required.

1.19.1 Optional: Daily Journalist Alerts

When a SecureDrop has little activity and receives only a few submissions every other week, checking daily only to find there is nothing is a burden. It is more convenient for *Journalists* to be notified daily via encrypted email about whether or not there has been submission activity in the past 24 hours.

If the email shows submissions were received, the *Journalist* can check their *Journalist Workstation*.

Note

For security reasons, the email will be sent every 24 hours, regardless of whether there are new submissions or not. The notification is sent after the daily reboot of the *Application Server*. The subject of the email will always be "Submissions in the past 24h". To find out whether there were submissions or not, a *Journalist* must decrypt the contents of the email.

In the simplest case a *Journalist* will provides their email and GPG public key to you, the admin. If a team of *Journalist* wants to receive these daily alerts, they should share a GPG key and ask the admin to setup a mail alias (SecureDrop does not provide that service) so they all receive the alerts and are able to decrypt them.

It is not possible to specify multiple email addresses for email notifications. If there are multiple intended recipients, use an alias or mailing list. However, all subscribers must share the GPG private key, as it is not possible to specify multiple keys.

If you wish to enable this, you will need:

- the email address that will receive the *Journalist* alerts
- the *Journalist Alert Public Key*
- the *Journalist Alert Public Key* fingerprint

Daily Journalist Alerts can be configured during or after installation.

1.19.2 OSSEC alerts

OSSEC is an open source host-based intrusion detection system (IDS) that SecureDrop uses to perform log analysis, file integrity checking, policy monitoring, rootkit detection, and real-time alerting. It is installed on the *Monitor Server* and constitutes that machine's main function. OSSEC works in a server-agent scheme; that is, the OSSEC server extends its existing functions to the *Application Server* through an agent installed on that server, covering monitoring for both machines.

In order to receive email alerts from OSSEC, you need to supply several settings during the SecureDrop server installation:

- The email address that will receive alerts from OSSEC
- The *OSSEC Alert Public Key* and its fingerprint
- The reachable hostname of your SMTP relay
- The secure SMTP port of your SMTP relay (typically 25, 587, or 465; must support TLS encryption)
- An email username to authenticate to the SMTP relay
- The domain name of the email used to send OSSEC alerts
- The password of the email used to send OSSEC alerts

Email address and public key

You must specify the email and GPG public key that you'll be using to receive alerts and decrypt the alert emails. You can use a pre-existing email and GPG key or create a new one specifically for receiving these alerts.

This could be your work email, or an alias for a group of IT admins at your organization. It helps for your mail client to have the ability to filter the numerous messages from OSSEC into a separate folder.

SMTP relay

Receiving email alerts from OSSEC requires that you have an SMTP relay to route the emails. You can use an SMTP relay hosted internally, if one is available to you, or you can use a *third-party SMTP relay such as Gmail*. The SMTP relay does not have to be on the same domain as the destination email address, i.e. smtp.gmail.com can be the SMTP relay and the destination address can be securedrop@freedom.press.

While there are risks involved with receiving these alerts, such as information leakage through metadata, we feel the benefit of knowing how the SecureDrop servers are functioning is worth it. If a third-party SMTP relay is used, that relay will be able to learn information such as the IP address the alerts were sent from, the subject of the alerts, and the destination email address the alerts were sent to. Only the body of an alert email is encrypted with the recipient's GPG key. A third-party SMTP relay could also prevent you from receiving any or specific alerts.

The SMTP relay that you use should support SASL authentication and SMTP TLS protocols TLSv1.2, TLSv1.1, and TLSv1. Most enterprise email solutions should be able to meet those requirements.

The SMTP relay mail server hostname is often, but not always, different from the SASL domain, e.g. smtp.gmail.com and gmail.com.

The SMTP and SASL settings correspond to the *outgoing* email address used to send the alerts instead of where you're receiving them. If that email is ossec@news-org.com, the SASL Username would be `ossec` and the SASL Domain would be `news-org.com`.

The settings and credentials for your SMTP relay must be provided during the SecureDrop server installation. It is better to get these right the first time rather than changing them after SecureDrop is installed. If you're not sure of the correct SMTP relay port number, you can use a simple mail client such as Thunderbird to test different settings or a port scanning tool such as nmap to see what's open. You could also use telnet to make sure you can connect to an SMTP server, which will always transmit a reply code of 220 meaning "Service ready" upon a successful connection.

In some cases, authentication or transport encryption mechanisms will vary and you may require later edits to the Postfix configuration (mainly /etc/postfix/main.cf) on the *Monitor Server* in order to get alerts to work. You can consult Postfix's official documentation for help, although we've described some common scenarios in the *troubleshooting section*.

Using Gmail for OSSEC alerts

It's possible SecureDrop to use Google's servers to deliver the alerts, but it's not ideal from a security perspective. This option should be regarded as a backup plan. Keep in mind that you're leaking metadata about the timing of alerts to a third party — the alerts are encrypted and only readable to you, however that timing may prove useful to an attacker.

First you should sign up for a new account. While it's technically possible to use an existing Gmail account, it's best to compartmentalize these alerts from any of your other activities. Choose a strong and random passphrase for the new account.

Next, enable Google's 2-Step Verification. This is required in order to use SMTP with a username and password, which is needed for SecureDrop.

After enabling 2-Step Verification, you'll then need to generate a new app password to use exclusively with SecureDrop. To do so, open the app password settings. From there, click "Select App", choose "Custom", assign it a name (such as "SecureDrop"), then click "Generate."

This will provide you with a 16-character password that you will need to use for the SMTP settings to enable OSSEC alerts.

Tip

SMTP through Gmail will only work with a generated app password. The password for the Gmail account itself is not sufficient, and will not allow mail to be sent. In order to be able to create an app password, you must have 2-Step Verification enabled on the Gmail account.

Once the account is created you can log out and use the SASL username as your new Gmail username (without the domain), the SASL domain to be either gmail.com or your custom Google Apps domain, and then finally your SASL password when installing SecureDrop on the servers. Remember to use the app password generated from the 2-step config, as the primary account password won't work. The SMTP relay will be smtp.gmail.com and the SMTP relay port is 587.

1.20 Prepare a SecureDrop Workstation

1.20.1 Overview

SecureDrop Workstation must be installed on a system running Qubes OS. The installation and configuration process should take between 4 and 6 hours, including time spent waiting for downloads and updates. At a high level, the tasks to be performed are as follows:

1.20.2 Prerequisites

In order to install SecureDrop Workstation and configure it to use an existing SecureDrop instance, you will need the following:

- A Qubes-compatible laptop based on the *hardware* recommendations.
- Qubes installation medium - this guide assumes the use of a USB 3.0 flash drive. Qubes may also be installed via optical media, which may make more sense depending on your security concerns.

> **Note**
>
> A USB flash drive with a Type-A connector is recommended, as USB-C ports may be disabled on your computer when the BIOS settings detailed below are applied.

- A working computer (Linux is recommended and assumed in this guide) to use for verification and creation of the Qubes installation medium.

> **Note**
>
> Tails can be used to perform the tasks below, but due to the size of the Qubes installation ISO, it may make sense to download it on another computer rather than via Tor, and then to use a USB flash drive to transfer it to Tails for verification and creation of the installation medium.

- A password manager or other system to generate and store strong passphrases for Qubes full disk encryption (FDE) and user accounts.

A basic knowledge of the Qubes OS is helpful.

1.20.3 Pre-install tasks

Apply BIOS updates and check settings

Before beginning the Qubes installation, make sure that your Qubes-compatible computer's BIOS is updated to the latest available version. For more details about this process, see the section on *Automatic BIOS updates*.

Once the BIOS is up-to-date, boot into the BIOS setup utility and update its settings. Note that not all BIOS versions will support the items listed, but if available following changes are recommended:

- Ensure the internal clock is correct.
- Set a password to access the BIOS (and record the password in your password manager).
- Disable BIOS downgrades.
- Enable Data Execution Prevention.
- Enable virtualization support (required for Qubes OS). - for Intel-based devices, **Intel VT-d** and **Intel VT-x** should be enabled - for AMD-based devices, **AMD-VI** and **AMD-V** should be enabled
- Disable unnecessary I/O options such as Wireless WAN and Bluetooth.
- Disable unnecessary network options such as Wake-on-LAN and UEFI network stacks.
- Disable Thunderbolt ports, or any other ports that allow Direct Memory Access (DMA).
- Enable any physical tamper detection options.
- Disable Computrace.
- Disable SecureBoot.

If the Qubes hardware compatibility list entry for your computer recommends the use of Legacy Mode for boot, change that setting in the BIOS as well.

Disable SecureBoot

SecureBoot is a feature available on most systems that, when enabled, does not allow any operating system to boot that has not been signed by a trusted key. By only booting to operating systems that are properly signed, you can be sure that the OS itself has not been corrupted or tampered with, at least at the boot level.

SecureBoot must be disabled on the server and Workstation hardware. SecureDrop installs a hardened, security-focused version of the Linux kernel (grsec) that does not support SecureBoot. If SecureBoot is enabled on either of the servers during the install, you will receive a pre-install error reminding you that it must be turned off before the installation can proceed.

Likewise, SecureBoot is not fully supported by QubesOS, and cannot be used with *SecureDrop Workstations*.

For instructions on how to enable or disable the SecureBoot feature for your device, please consult the manufacturer's manual for BIOS settings, as they differ for each make and model.

Download and verify Qubes OS

On the working computer, download the Qubes OS ISO and cryptographic hash values for version `4.2.4` from https://www.qubes-os.org/downloads/. The ISO is 6.8 GB approximately, and may take some time to download based on the speed of your Internet connection.

Follow the linked instructions to verify the ISO. Ensure that the ISO and hash values are in the same directory, then run:

```
gpg --keyserver-options no-self-sigs-only,no-import-clean --fetch-keys https://keys.
↪qubes-os.org/keys/qubes-release-4.2-signing-key.asc
gpg -v --verify Qubes-R4.2.4-x86_64.iso.DIGESTS
sha256sum -c Qubes-R4.2.4-x86_64.iso.DIGESTS
```

The output should look like this:

```
gpg: requesting key from 'https://keys.qubes-os.org/keys/qubes-release-4.2-signing-key.
↪asc'
gpg: key E022E58F8E34D89F: public key "Qubes OS Release 4.2 Signing Key" imported
gpg: Total number processed: 1
gpg:               imported: 1
gpg: no ultimately trusted keys found

gpg: armor header: Hash: SHA256
gpg: original file name=''
gpg: Signature made Mon 17 Feb 2025 12:00:00 AM EST
gpg:                using RSA key 9C884DF3F81064A569A4A9FAE022E58F8E34D89F
gpg: using pgp trust model
gpg: Good signature from "Qubes OS Release 4.2 Signing Key" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 9C88 4DF3 F810 64A5 69A4  A9FA E022 E58F 8E34 D89F
gpg: textmode signature, digest algorithm SHA256, key algorithm rsa4096
Qubes-R4.2.4-x86_64.iso: OK
sha256sum: WARNING: 20 lines are improperly formatted
```

Specifically, you will want to make sure that you see "Good signature" listed in the text. If it does not report a good signature, try deleting the ISO and downloading it again.

Once you've verified the ISO, copy it to your installation medium - for example, if using Linux and a USB flash drive, using the command:

```
sudo dd if=Qubes-R4.2.4-x86_64.iso of=/dev/sdX bs=1048576 && sync
```

where `if` is set to the path to your downloaded ISO file and `of` is set to the block device corresponding to your USB flash drive. Note that any data on the USB flash drive will be overwritten.

Caution

Make sure to verify that you have the correct device name using, for example, the `lsblk` command. You should write to the full device (eg. `/dev/sdc`) rather than to a partition (eg. `/dev/sdc1`).

Install Qubes OS (estimated wait time: 30-45 minutes)

Before starting the installation, please ensure that:

- the computer is charging
- all USB devices like YubiKeys, mice and keyboards are disconnected

To begin the Qubes installation, connect the Qubes installation drive you just created to your target computer and boot from it. You may need to bring up a boot menu at startup to do so - on Lenovo laptops, for example, you can do so by pressing **F12** on boot.

Follow the installation documentation to install Qubes on your computer, ensuring that you:

- Use English - United States as the setup language. (This requirement will be dropped in a future version).
- Use all available storage space for the installation (as the computer should be dedicated to SecureDrop Workstation).
- Set a strong full disk encryption (FDE) passphrase - a 6-word Diceware passphrase is recommended.
- Create an administrative account named `user` with a strong password.

Note

Qubes is not intended to have multiple user accounts, so your account name and password will be shared by all SecureDrop Workstation users. The password will be required to log in and unlock the screen during sessions - choosing something strong but memorable and easily typed is recommended!

Once the installation is complete, you will be prompted to reboot into Qubes. Reboot, removing the install USB flash drive when the computer restarts.

You will be prompted to enter the FDE passphrase set during installation.

After the disk is unlocked and Qubes starts, you will be prompted to complete the initial setup. Click the Qubes OS icon.

On the configuration screen, ensure that the following options are checked:

- Default Template should be set to "Fedora 41 Xfce"
- "Create default system qubes (sys-net, sys-firewall, default DispVM)"
- "Make sys-firewall and sys-usb disposable"

If there is a grayed out option "USB qube configuration disabled", make a note of this. An additional setup step will be required (see next section).

Finally, click **Finish Configuration** to set up the default system TemplateVMs and AppVMs.

Once the initial setup is complete, the login dialog will be displayed. Log in using the username and password set during installation.

(Hardware-dependent) Apply USB fixes

If, during the installation, you encountered the grayed out option "USB qube configuration disabled", you must now create a VM to access your USB devices. If you did not encounter this issue, you can skip this section.

To create a USB qube, open a `dom0` terminal via  ▶  ▶ **Other Tools** ▶ **Xfce Terminal**.

Tip

For quicker access, you can add the `dom0` terminal to the "Favorites" section of the Qubes menu (identified by a bookmark symbol). Right-click the entry and select **Add to favorites**. To remove it at a later time, right-click the entry in your list of favorites and select **Remove from favorites**.

Run the following command:

```
sudo qubesctl state.sls qvm.sys-usb
```

After the command exits, confirm that you see an entry "Service: sys-usb" in the Qubes menu. If `sys-usb` is not running, you can start it with the command `qvm-start sys-usb` in `dom0`. Once `sys-usb` is running, click the devices widget in the upper right panel to expand a listing of all devices detected by Qubes OS.

Now, insert a safe USB device you intend to use with the SecureDrop Workstation. Click the devices widget again. Does the newly attached USB device appear in the list? If so, USB support is working and you can proceed with the installation. If you do encounter the error message "Denied qubes.InputKeyboard from sys-usb to dom0", you need to additionally enable USB keyboard support:

```
sudo qubesctl state.sls qvm.usb-keyboard
```

While we recommend against the use of a USB keyboard for security reasons, this error can also occur in combination with other USB devices on some hardware.

Apply `dom0` updates (estimated wait time: 15-30 minutes)

`dom0` is the most trusted domain on Qubes OS, and has privileged access to all other VMs. As such, it is important to ensure that all available security updates have been applied to `dom0` as the first step after the installation.

After logging in, use the network manager widget in the upper-right panel to configure your network connection.

Open a `dom0` terminal from the Qubes Application menu (the  icon in the upper left corner) by selecting  ▶  **(left-hand side)** ▶ **Other Tools** ▶ **Xfce Terminal**. Run the following command:

```
sudo qubes-dom0-update -y
```

Wait for all updates to complete. If you encounter an error during this stage, please contact us for assistance, as it may not be safe to proceed with the installation.

After updating `dom0`, reboot the workstation to ensure that all updates have taken effect for your active session.

Apply updates to system templates (estimated wait time: 45-60 minutes)

After logging in again, confirm that the network manager successfully connects you to the configured network. If necessary, verify the network settings using the network manager widget.

- Next, configure Tor via  ▶ **Service** ▶ **sys-whonix** ▶ **Anon Connection Wizard**. In most cases, choosing the default **Connect** option is best. Click **Next**, then **Next** again. Then, if Tor connects successfully, click **Finish**. If Tor fails to connect, make sure your network conection is up and does not filter Tor connections, then try again.

  Note

  If Tor connections are blocked on your network, you may need to configure Tor to use bridges in order to get a connection. For more information, see the Anon Connection Wizard documentation.

- Once Tor has connected, launch the Qubes Update tool via  ▶ **Qubes Tools** ▶ **Qubes Update** to update the system VMs. in the `[Dom0] Qubes Update` window, check all entries in the list above except for `dom0` (which you have already updated in the previous step). Then, click **Update**. The system's VMs will be updated sequentially - this may take some time. When the updates are complete, click **Next**. You will then be prompted to **Finish and restart/shutdown 4 qubes.** Go ahead and do so, and allow time for them to restart.

1.20.4 Installing SecureDrop Workstation

Download SecureDrop Workstation packages

First, you must configure the Qubes-Contrib repo, then download the SecureDrop Workstation packages.

- Make sure that network connection is enabled using the network manager widget in the upper right panel.
- Next, in a `dom0` terminal ( ▶  ▶ **Other** ▶ **Xfce Terminal**):

```
sudo qubes-dom0-update -y qubes-repo-contrib
sudo qubes-dom0-update --clean -y securedrop-workstation-keyring
```

- The SecureDrop Relase keyring will be installed on your machine. Wait 15 seconds for the key to be imported into the `rpm` database. Then:

```
sudo qubes-dom0-update --clean -y securedrop-workstation-dom0-config
sudo dnf -y remove qubes-repo-contrib
```

1.20.5 Install *securedrop-admin* tooling

1.20.6 Generate *Submission Private Key*

1.21 Generate the *Submission Key*

When a document or message is submitted to SecureDrop by a *Source*, it is automatically encrypted with the *Submission Key*. The private part of this key is only stored on the *Secure Viewing Station* which is never connected to the Internet. SecureDrop submissions can only be decrypted and read on the *Secure Viewing Station*.

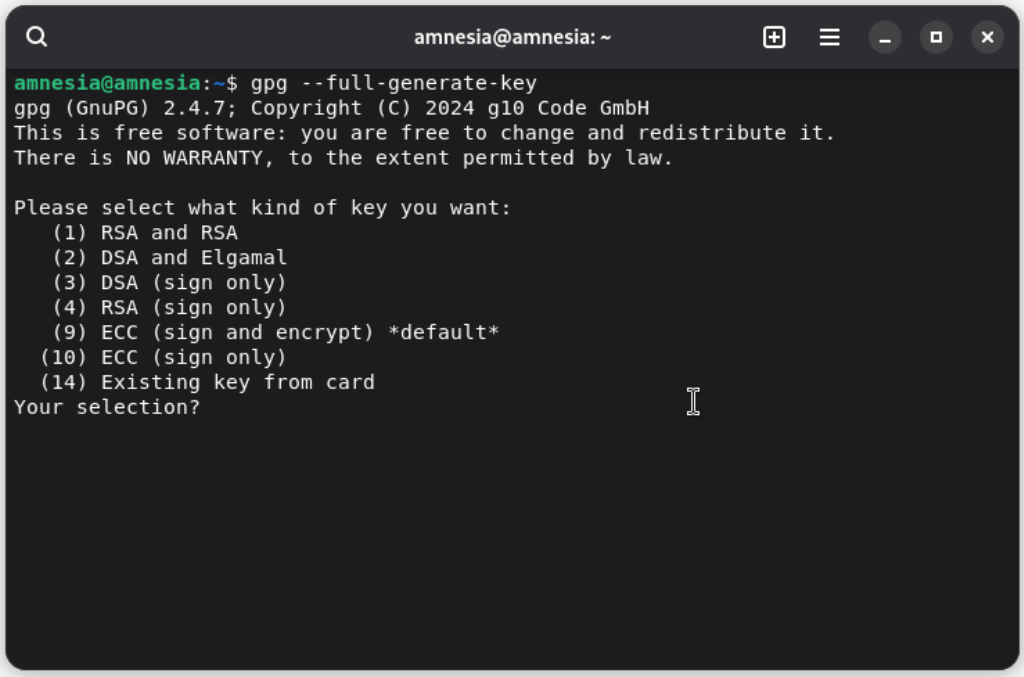
We will now generate the *Submission Key*. If you aren't still logged into your *Secure Viewing Station* from the previous step, boot it using its Tails USB flash drive, with persistence enabled.

Important

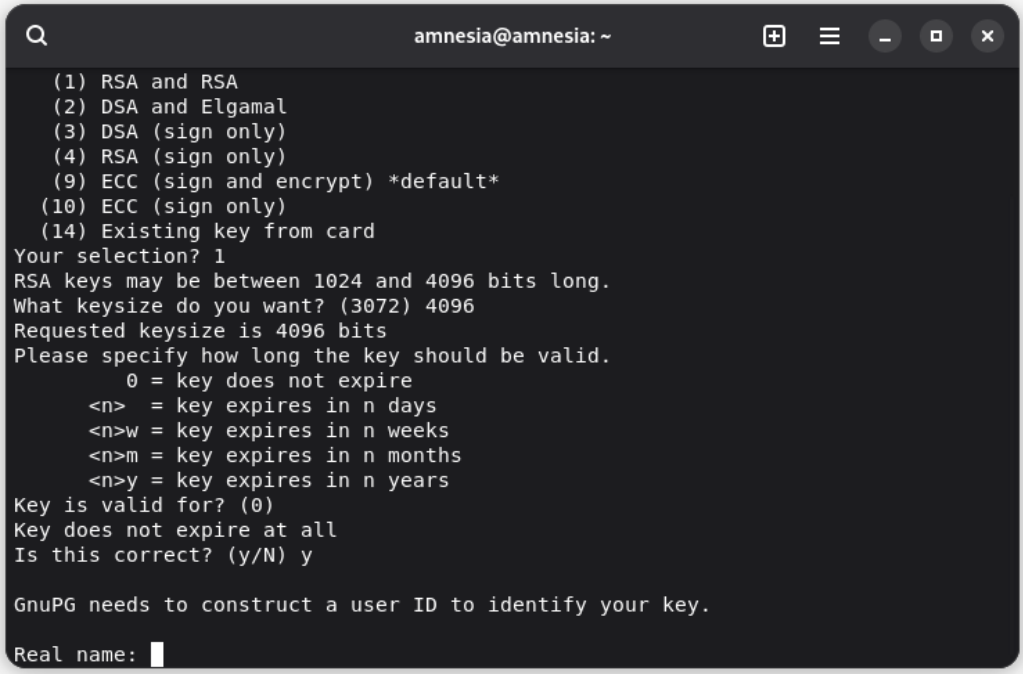
The private key you will generate in the following steps is one of the most important secrets associated with your SecureDrop installation. This procedure is intended to ensure that the private key is protected by the air-gap throughout its lifetime.

1.21.1 Create the key

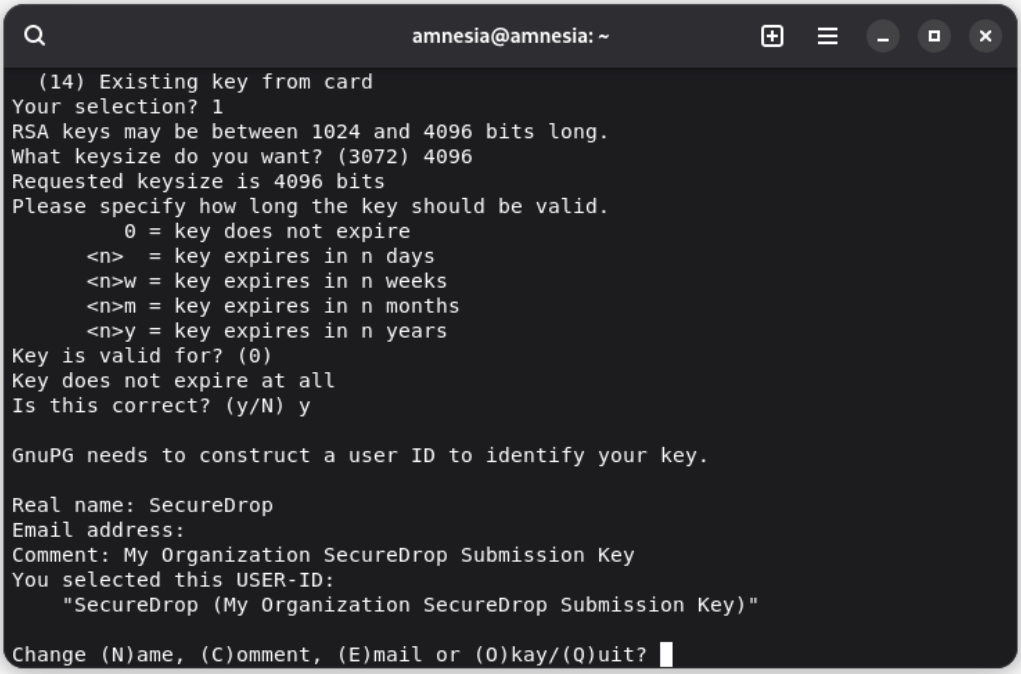
1. Navigate to **Apps ▶ System Tools ▶ Console** to open a terminal  .
2. In the terminal, run `gpg --full-generate-key`:



3. When it says **Please select what kind of key you want**, choose "*(1) RSA and RSA (default)*".
4. When it asks **What keysize do you want?**, type `4096`.
5. When it asks **Key is valid for?**, press Enter. This means your key does not expire.
6. It will let you know that this means the key does not expire at all and ask for confirmation. Type **y** and hit Enter to confirm.



7. **Next it will prompt you for user ID setup. Use the following options:**
    - **Real name**: "SecureDrop"
    - **Email address**: leave this field blank
    - **Comment**: `[Your Organization's Name] SecureDrop Submission Key`
8. GPG will confirm these options. Verify that everything is written correctly. Then type `0` for `(0)kay` and hit enter to continue:

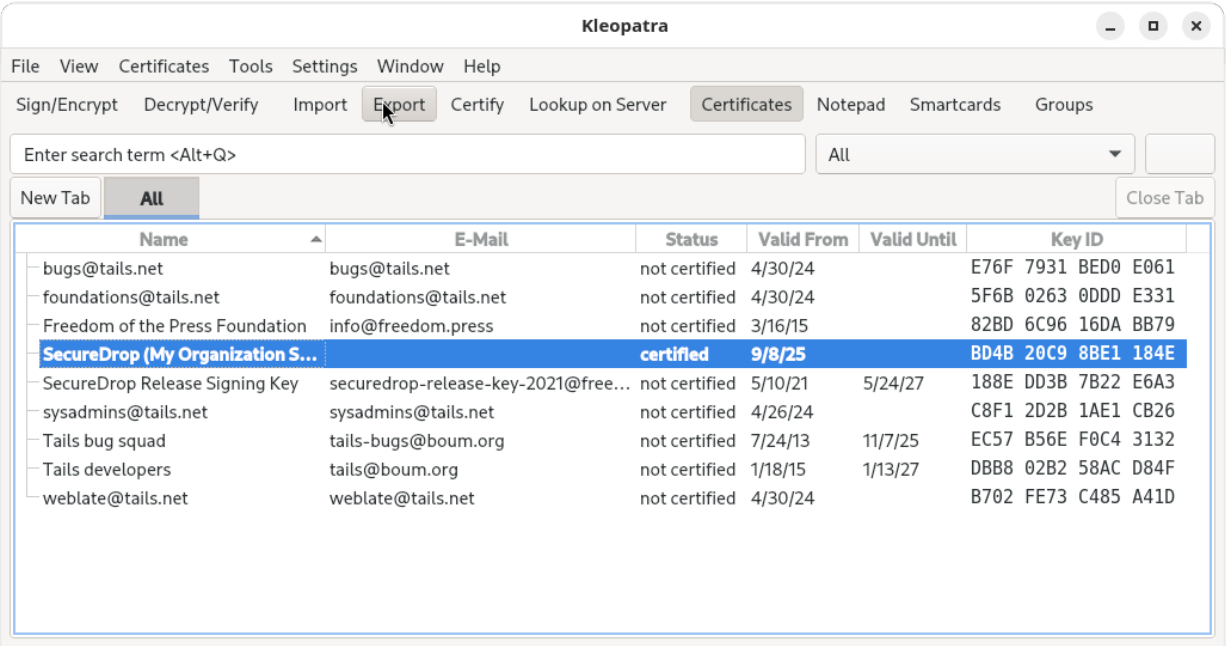


9. A box will pop up (twice) asking you to type a passphrase. Since the key is protected by the encryption on the Tails persistent volume, it is safe to simply click **OK** without entering a passphrase.
10. The software will ask you if you are sure. Click **Yes, protection is not needed**.
11. Wait for the key to finish generating.

1.21.2 Export the *Submission Public Key*

Navigate to **Apps ► Accessories ► Kleopatra** to open a graphical interface to manage GPG keys. Once Kleopatra opens you will find a list of keys, including the SecureDrop *Submission Key* you just created.

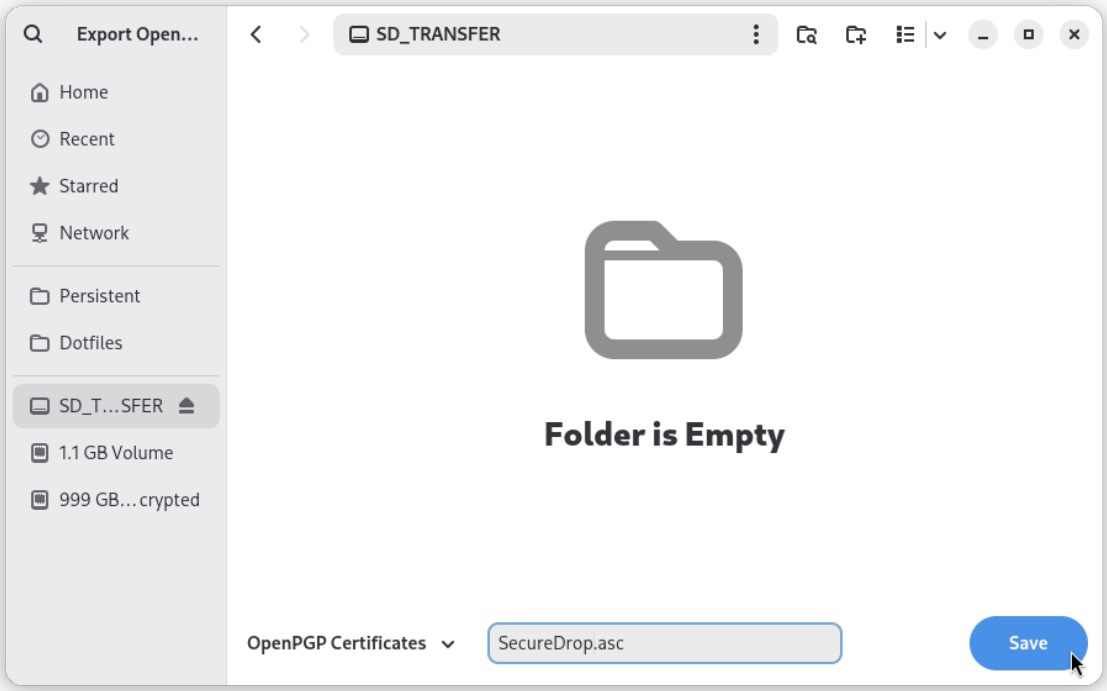
Click to select the key, then click the "Export…" button in the toolbar above.



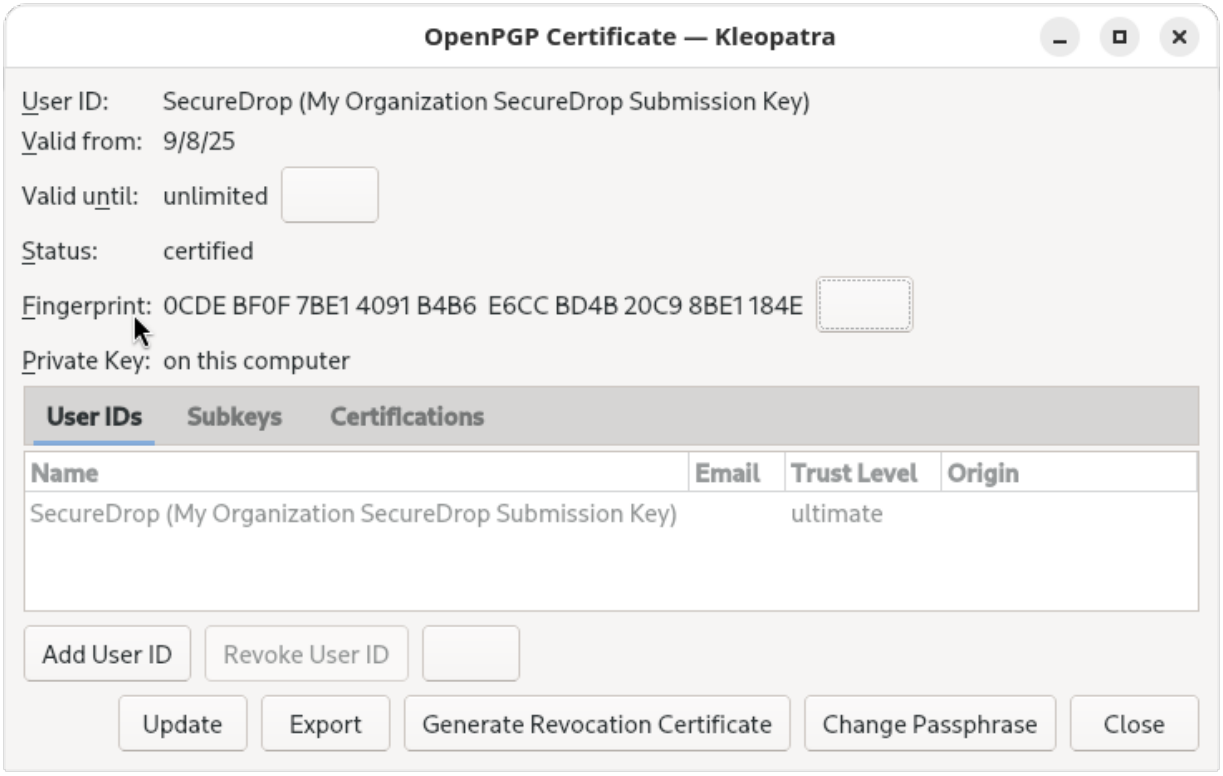
Save the key to the *Transfer Device* by changing the location to `/media/amnesia/Transfer Device`, then set the filename to `SecureDrop.asc`. Once that is set, click the *Save* button to finish exporting the key to the transfer device.

Note

This is the public key only.



After exporting the public key, you will be returned back to the list of keys. You'll need to provide the fingerprint of the *Submission Key* during the installation. Go ahead and double-click on the *Submission Key*, then write down the 40 hexadecimal digits under *Fingerprint*.



Note

Your fingerprint will be different from the one in the example screenshot.

At this point, you are done with the *Secure Viewing Station* for now. You can shut down Tails, grab the *Admin Workstation* USB flash drive, and move over to your regular workstation.

1.22 Using the KeePassXC password manager

Qubes OS comes with the KeePassXC password manager preinstalled. As outlined in our *passphrase best practices*, we recommend all SecureDrop users, including administrators, use the KeePassXC password manager to generate and retain strong and unique passphrases.

1.22.1 Template password database

To facilitate using KeePassXC to organize all the credentials needed for using and administering the system, SecureDrop comes with a template database with organized entries ready to be filled in.

- Open the KeePassXC program in the *vault* VM
- Select **Database ▶ Open database**, and navigate to the location of **/path/to/Passwords.kdbx**, select it, and click **Open**
- Leave the password blank and click **OK**. If you receive an "Unlock failed" prompt, click **Retry with empty password**.
- Edit entries as required.
- Select **Database ▶ Save Database** to save your changes.

The next time you use KeepassXC in *vault*, the database at `/path/to/Passwords.kdbx` will be selected by default.

KeePassXC will show a warning every time you attempt to open a database without entering a password. Because your persistent volume is encrypted, setting up this additional password is not strictly required. It provides some additional protection, e.g., if a computer is left running, at the cost of convenience.

For passwordless access without warnings, you can protect the database using a key file, via **Database ▶ Database settings ▶ Security ▶ Add additional protection ▶ Add Key File ▶ Generate**. This key file has to be stored in your Persistent folder and it must be selected when you open the database.

After configuring the password database, restart KeePassXC once to verify that you are able to access it as expected.

Warning

You will not be able to access your passwords if you forget the full disk encryption or the location of the key file used to protect the database.

In case you wish to manually create a database, the suggested password fields in the template are:

Admin:

- Admin account username
- *Application Server* SSH Onion address
- Email account for sending OSSEC alerts
- *Monitor Server* SSH Onion address
- Network Firewall Admin Credentials
- *OSSEC Alert Public Key*
- SecureDrop Login Credentials

Journalist:

- Auth Value: *Journalist Interface*
- Onion URL: *Journalist Interface*
- Personal GPG Key
- SecureDrop Login Credentials

Backup:

- This section contains clones of the above entries in case a user accidentally overwrites an entry.

As you proceed with the installation, enter the credentials you create in this database as you go.

1.23 Set up the network firewall

Now that you've set up your password manager, you can move on to setting up the Network Firewall. You should stay logged in to the *Admin Workstation* to access the Network Firewall's web interface for configuration.

Unfortunately, due to the wide variety of firewalls that may be used, we do not provide specific instructions to cover every type or variation in software or hardware. However, if you have the necessary expertise, we provide *abstract firewall rules* that can be implemented with iptables, Cisco IOS etc. We recommend that you use a firewall with at least four physical interfaces.

The documentation linked below describes the configuration procedure for pfSense- and OPNSense-based firewalls. One option not covered in this guide is to build your own network firewall and install OPNSense on it. However, for most installations, we recommend buying a dedicated firewall appliance with your firewall OS of choice pre-installed.

Please note that we no longer recommend the use of pfSense Community Edition (CE) due to changes in the frequency and scope of security updates made available there. pfSense Plus continues to receive necessary security updates on a regular basis, and is provided with the purchase of most Netgate firewalls.

We currently recommend three firewalls in our *Hardware Guide*:

- The Netgate SG-4100, a pfSense-based firewall with 6 network interfaces: 2 WAN ports and 4 LAN ports.
- The Netgate SG-6100, a pfSense-based firewall with 8 network interfaces: 4 WAN ports and 4 LAN ports.
- The Protectli Vault 4-Port (with coreboot), an OPNSense-based open-source hardware firewall with 4 configurable network interfaces.

1.23.1 Configuration: pfSense

If you are using a pfSense-based firewall such as the recommended SG-4100, follow the instructions to *Configure a pfSense firewall for use with SecureDrop*.

1.23.2 Configuration: OPNSense

If you are using an OPNSense-based firewall such as the recommended APu4D4, follow the instructions to *Configure an OPNSense firewall for use with SecureDrop*.

1.23.3 Configuration: other firewalls

If you are using a firewall based on an OS not listed above, you should still set it up use the same overall configuration and ruleset as defined for the supported models.

The *Application* and *Monitor Servers* should be set up on separate subnets configured on separate physical NICs, with the *Admin Workstation* also on a separate subnet if possible. Including the WAN connection, a minimum of 4 NICs must be available.

The abstract ruleset required by SecureDrop can be described as follows:

- Disable DHCP (in case the firewall is providing a DHCP server by default)
- Disallow all traffic by default (inbound or outbound)
- Allow UDP OSSEC (port 1514) from *Application Server* to *Monitor Server*
- Allow TCP ossec agent auth (port 1515) from *Application Server* to *Monitor Server*
- Allow TCP/UDP DNS from *Application Server* and *Monitor Server* to the IPs of known name servers
- Allow UDP NTP from *Application Server* and *Monitor Server* to all
- Allow TCP any port from *Application Server* and *Monitor Server* to all (this is needed for making connections to the Tor network)
- Allow TCP 80/443 from *Admin Workstation* to all (in case there is a need to access the web interface of the firewall)
- Allow TCP SSH from *Admin Workstation* to *Application Server* and *Monitor Server*
- Allow TCP any port from *Admin Workstation* to all

This can be implemented with iptables, Cisco IOS etc. if you have the necessary expertise.

1.24 Setting up a pfSense network firewall

1.24.1 Before you begin

First, consider how the firewall will be connected to the Internet. You will need to provision several unique subnets, which should not conflict with the network configuration on the WAN interface. If you are unsure, consult your local system administrator.

Many firewalls, including the recommended pfSense-based devices, automatically set up the LAN interface on `192.168.1.1/24`. This particular private network is also a very common choice for home and office routers. If you are connecting the firewall to a router with the same subnet (common in a small office, home, or testing environment), you will probably be unable to connect to the network at first. However, you will be able to connect from the LAN to the pfSense WebGUI configuration wizard, and from there you will be able to configure the network so it is working correctly.

Configuring your firewall

Since our recommended firewalls have at least 4 NICs, we will refer to the relevant ports as WAN[1], LAN[1], LAN2, and LAN3. (Bracketed numbers may be present on the physical ports' labels but not in the pfSense UI.) In this case, we can now use a dedicated port on the network firewall for each component of SecureDrop (*Application Server*, *Monitor Server*, and *Admin Workstation*).

Depending on your network configuration, you should define the IP and subnet values your instance will use before continuing. We recommend the default values below:

IP and subnet definitions:

- Admin Subnet: `10.20.1.0/24`
- Admin Gateway: `10.20.1.1`
- Admin Workstation (LAN[1]): `10.20.1.2`
- Application Subnet: `10.20.2.0/24`
- Application Gateway: `10.20.2.1`
- Application Server (LAN2): `10.20.2.2`
- Monitor Subnet: `10.20.3.0/24`
- Monitor Gateway: `10.20.3.1`
- Monitor Server (LAN3) : `10.20.3.2`

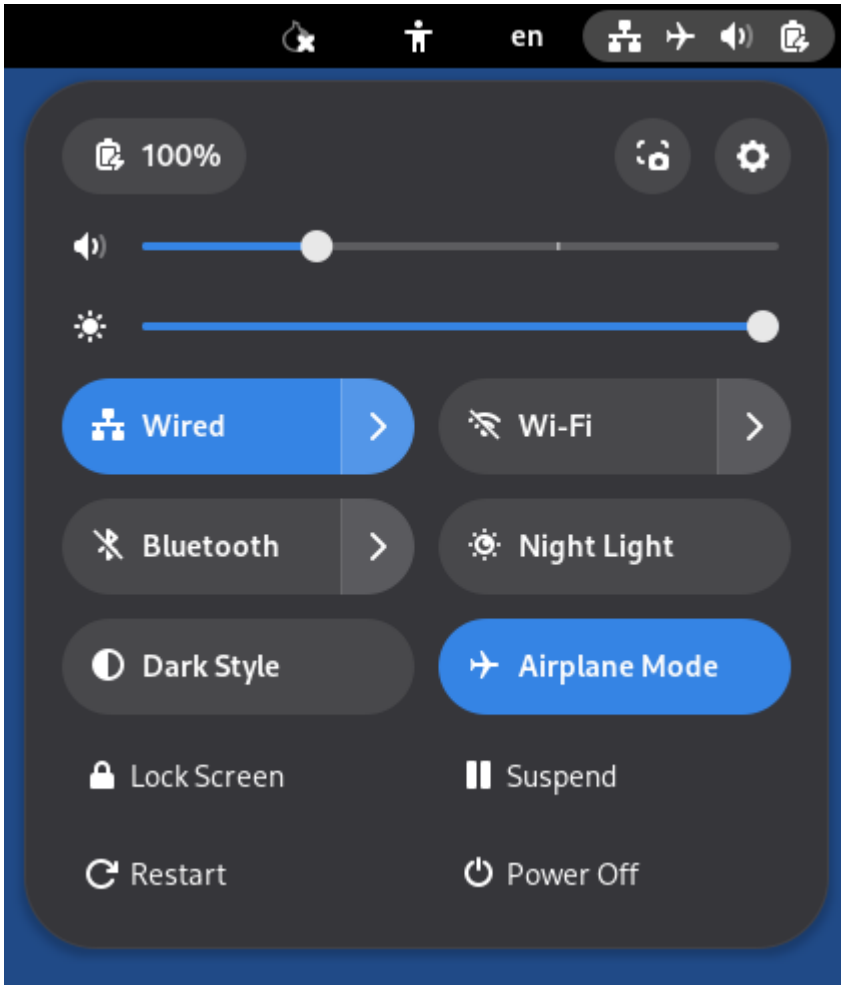
1.24.2 Initial configuration

Unpack the firewall, connect the power, and power on the device.

We will use the pfSense WebGUI to do the initial configuration of the network firewall.

Connect to the pfSense web GUI

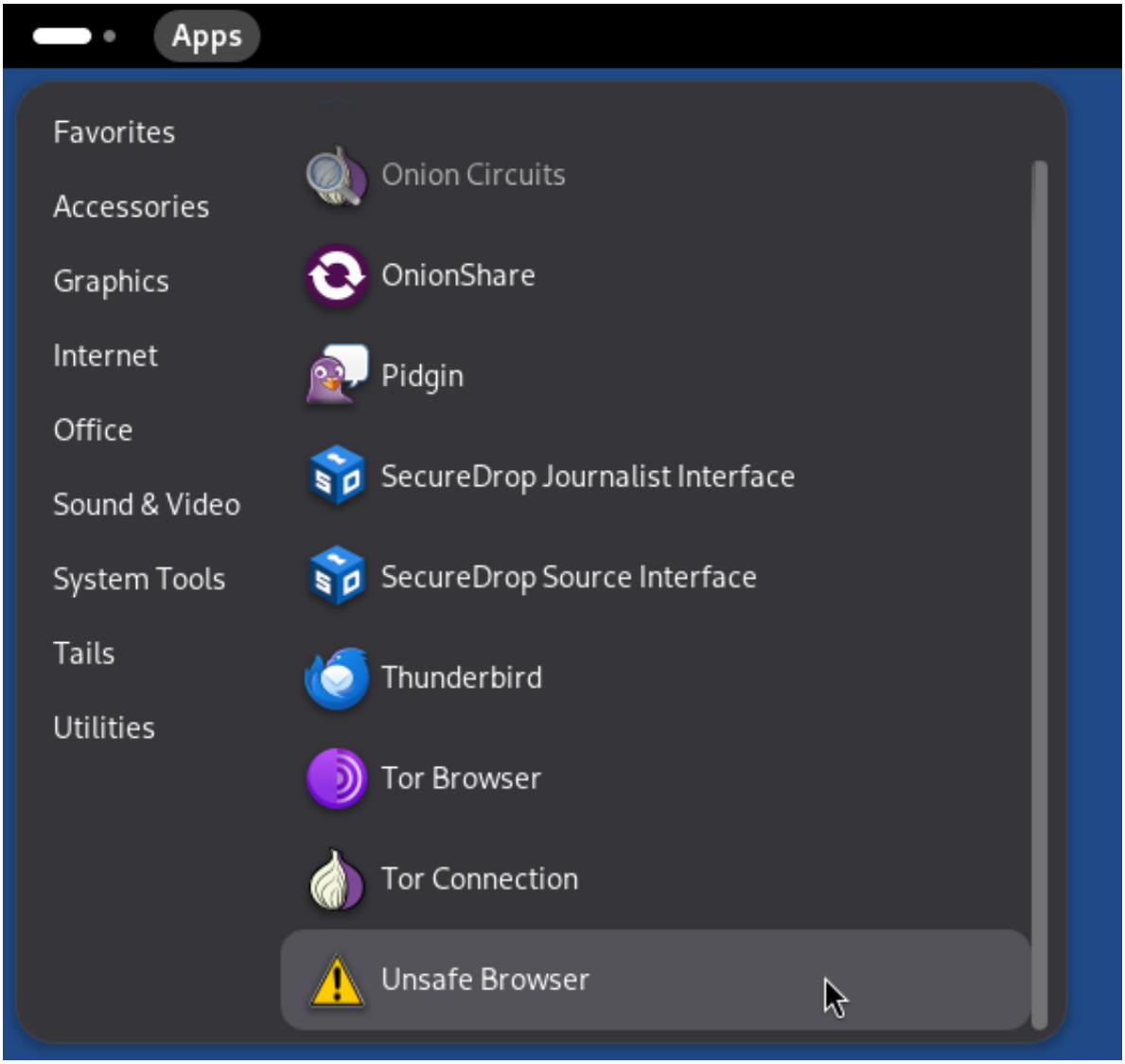
1. If you have not already done so, boot the *Admin Workstation*.
2. Connect the *Admin Workstation* to the LAN[1] interface. You should see a popup notification in Tails that says "Connection Established". If you click on the network icon in the upper right of the Tails Desktop, you should see that the Wired Connection is active:



Warning

Make sure your *only* active connection is the one you just established with the network firewall. If you are connected to another network at the same time (e.g. a wireless network), you may encounter problems trying to connect the pfSense WebGUI.

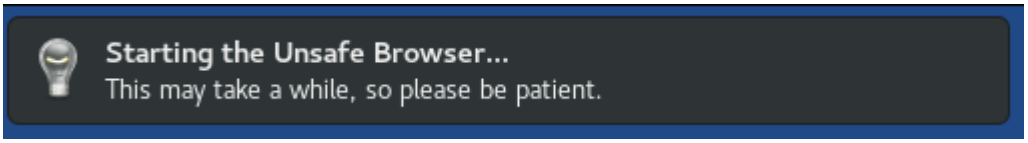
3. Launch the **Unsafe Browser** from the menu bar: **Apps ► Internet ► Unsafe Browser**.



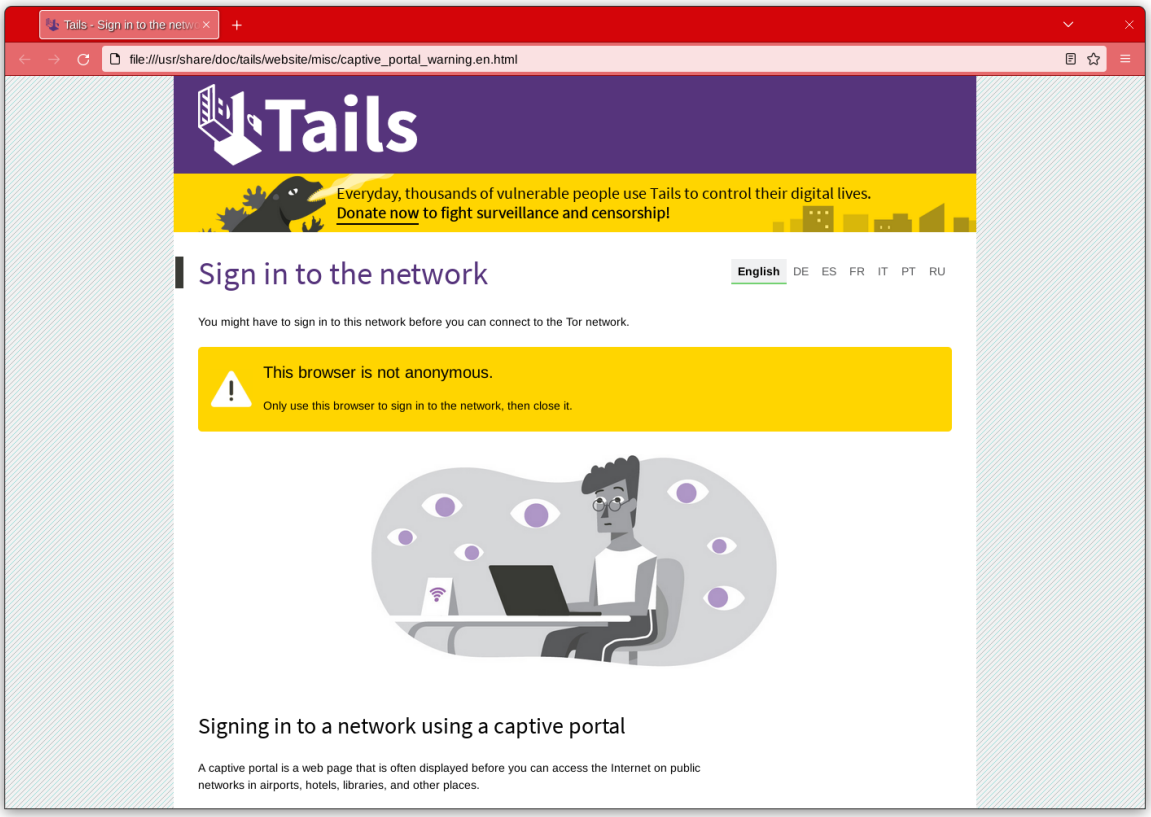
Note

The *Unsafe Browser* is, as the name suggests, **unsafe** (its traffic is not routed through Tor). However, it is the only option because Tails intentionally disables LAN access in the **Tor Browser**.

4. You will see a pop-up notification that says "Starting the Unsafe Browser…"



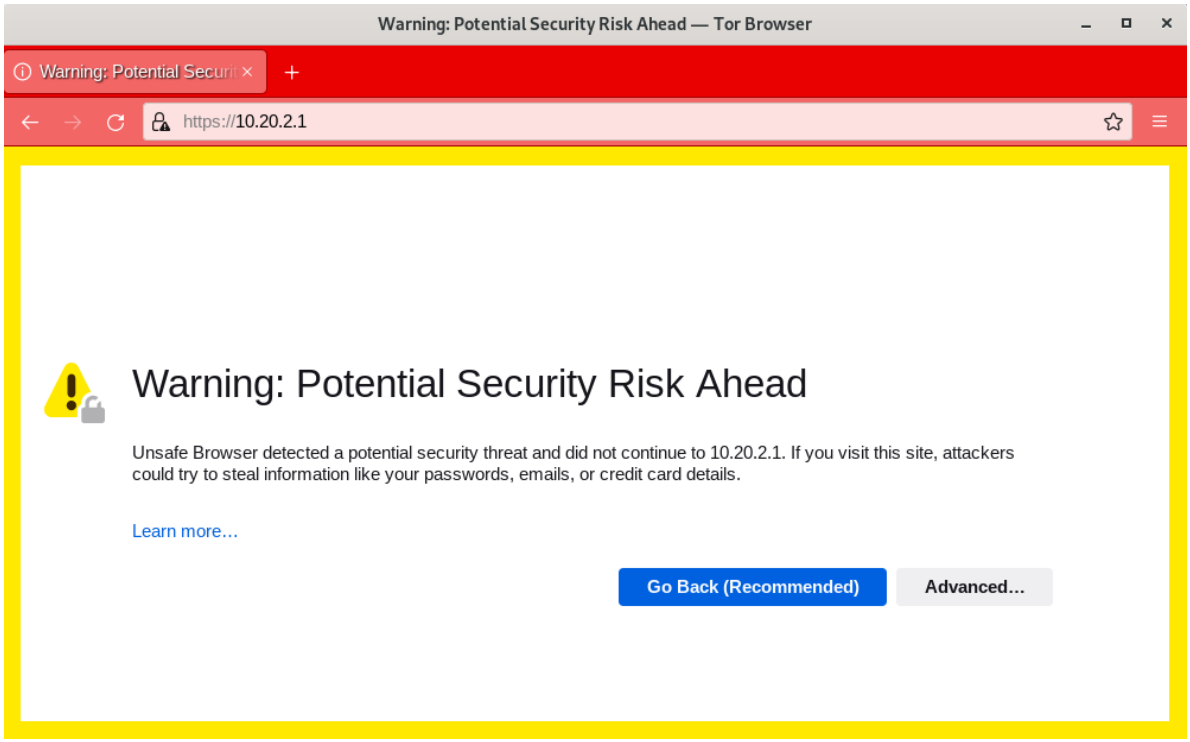
5. After a few seconds, the Unsafe Browser should launch. The window has a bright red border to remind you to be careful when using it. You should close it once you're done configuring the firewall and use Tor Browser for any other web browsing you might do on the *Admin Workstation*.



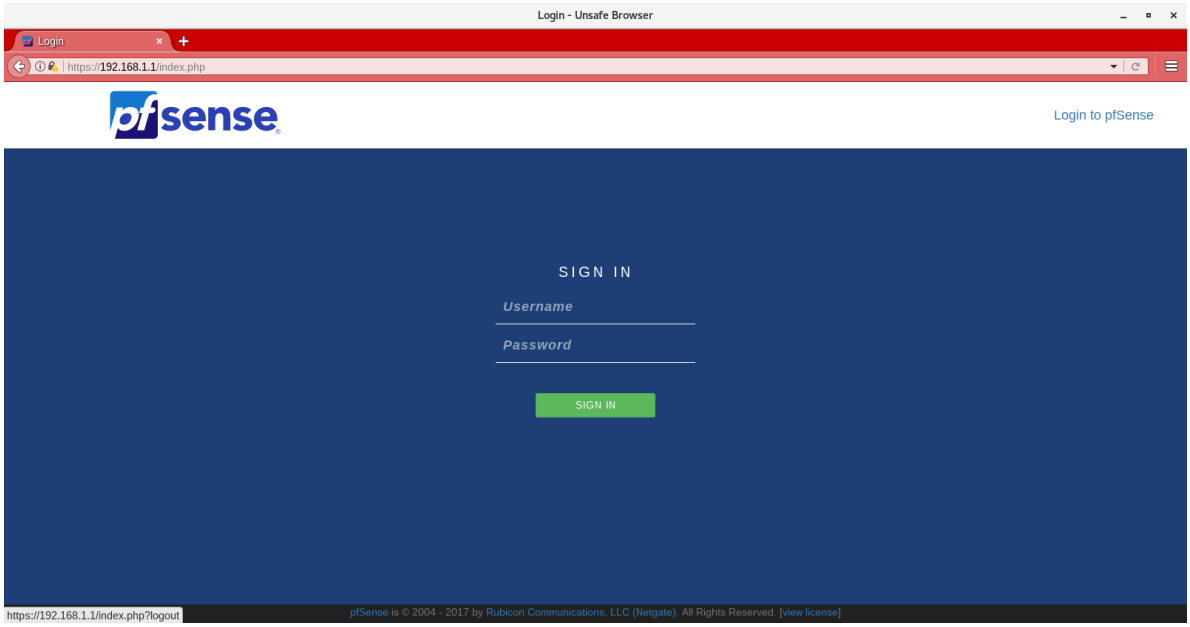
6. Navigate to the pfSense WebGUI in the *Unsafe Browser*: `https://192.168.1.1`

> **Note**
>
> If you have trouble connecting, go to your network settings and make sure that you have an IPv4 address in the `192.168.1.1/24` range. You may need to turn on DHCP, else you can manually configure a static IPv4 address of `192.168.1.x` with a subnet mask of `255.255.255.0`. However, make sure not to configure your Tails device to have the same IP as the firewall (`192.168.1.1`).

7. The firewall uses a self-signed certificate, so you will see a "Potential Security Risk Ahead" warning when you connect. This is expected. You can safely continue by clicking **Advanced**, then **Accept the Risk and Continue**.

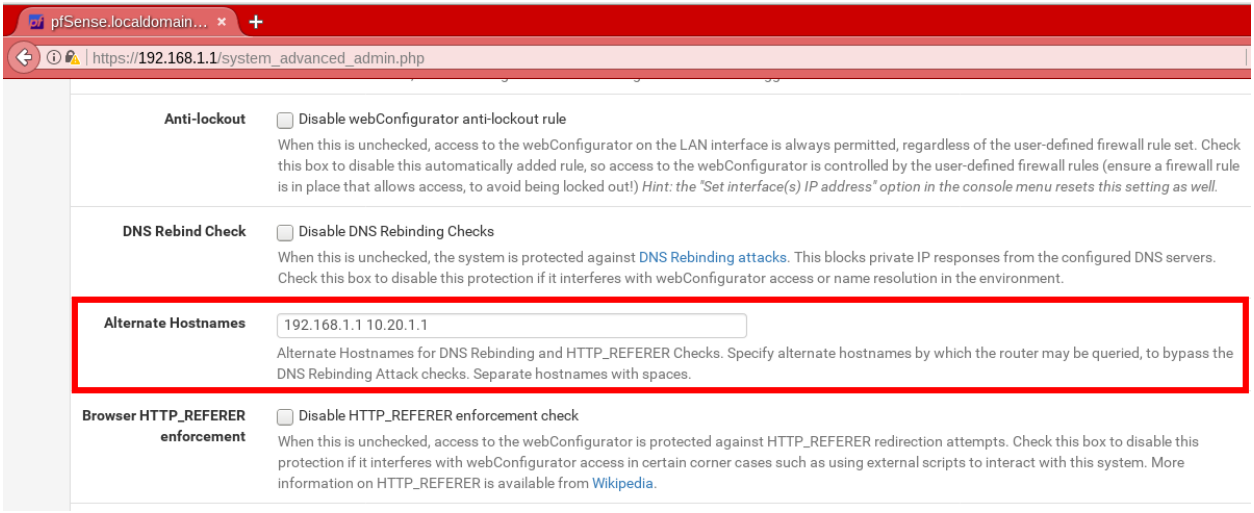


8. You should see the login page for the pfSense GUI. Log in with the default username and passphrase (`admin` / `pfsense`).



Alternate hostnames

Before you can set up the hardware firewall, you will need to set the **Alternate Hostnames** setting after logging in. You will see the Setup Wizard but you should exit out of it by navigating to **System ► Advanced**. In the **Alternate Hostnames** dialog box, add `192.168.1.1` as well as the IP address of the *Admin Gateway*. If you decide against using our recommended defaults for the *Admin Gateway*, you should include that value here. After saving these settings you should be able to go back to **System** and select **Setup Wizard**.

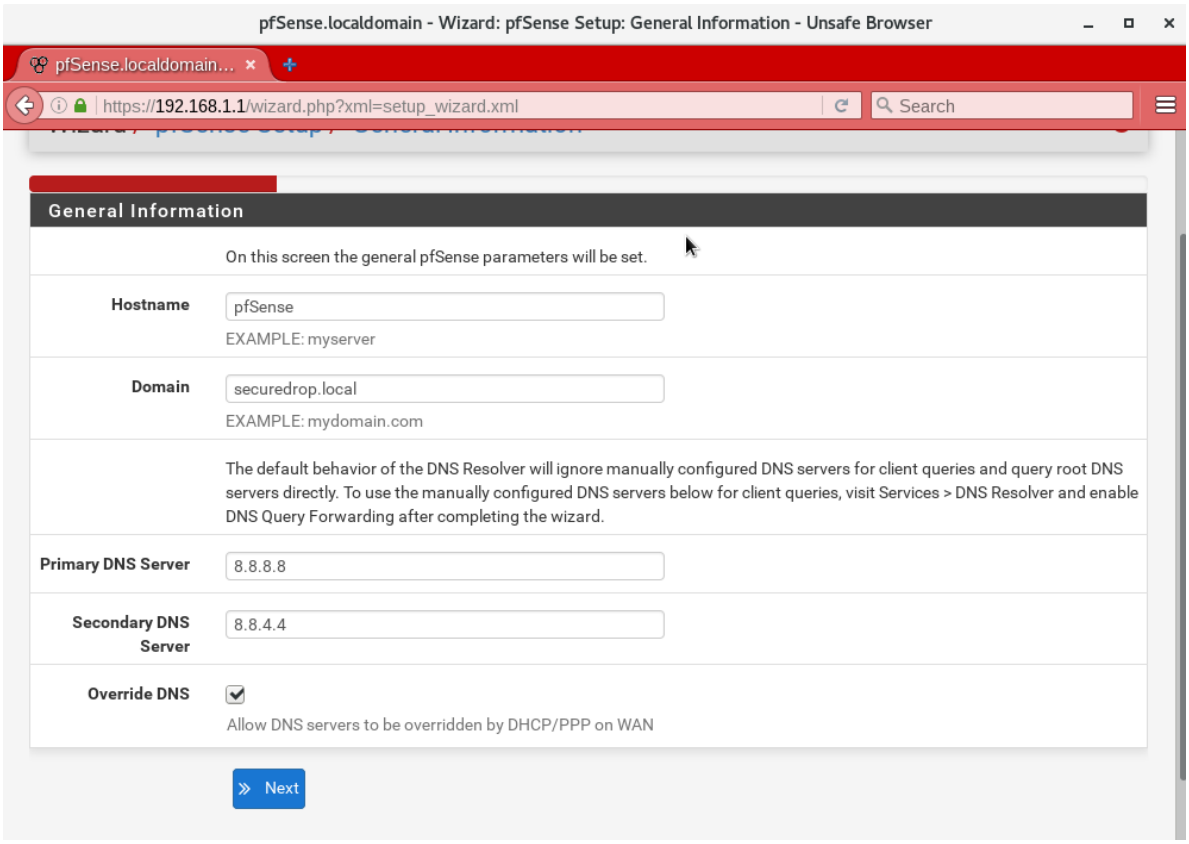


Note

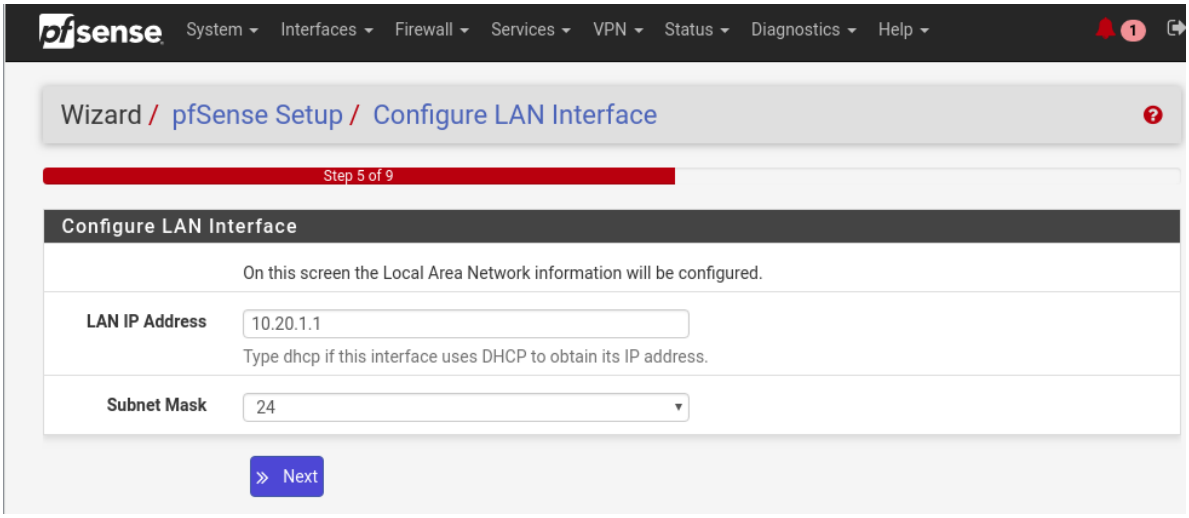
If you are using a different IP for the Admin Gateway you should enter that IP in the Alternate Hostname field. Failure to do so will result in an error with the text "An HTTP_REFERER was detected other than what is defined in System -> Advanced". If you see this error you may have to do a factory reset of the firewall via the serial console.

Setup Wizard

1. If you're setting up a brand new (or recently factory reset) router, logging in to the pfSense WebGUI will automatically start the Setup Wizard. Click **Next**, then **Next** again. Don't sign up for a pfSense Gold subscription (unless you want to).
2. On the "General Information" page, we recommend leaving your hostname as the default (pfSense). There is no relevant domain for SecureDrop, so we recommend setting this to `securedrop.local` or something similar. Use your preferred DNS servers. If you don't know what DNS servers to use, we recommend using Google's DNS servers: `8.8.8.8` and `8.8.4.4`. Click Next.



3. Leave the defaults for "Time Server Information". Click **Next**.
4. On "Configure WAN Interface", enter the appropriate configuration for your network. Consult your local sysadmin if you are unsure what to enter here. For many environments, the default of DHCP will work and the rest of the fields can be left blank. Click **Next**.
    - If your firewall is behind another firewall or NAT device, you will need to deselect the **Block private networks from entering via WAN** option to allow traffic to and from your upstream network.
5. For "Configure LAN Interface", use the IP address of the *Admin Gateway* (`10.20.1.1`) and the subnet mask (`/24`) of the *Admin Subnet*. Click **Next**.



6. Set a strong admin passphrase. We recommend generating a strong passphrase with KeePassXC, and saving it in the Tails Persistent folder using the provided KeePassXC database template. Click **Next**.
7. Click Reload. Once the reload completes and the web page refreshes, click the corresponding "here" link to "continue on to the pfSense webConfigurator".

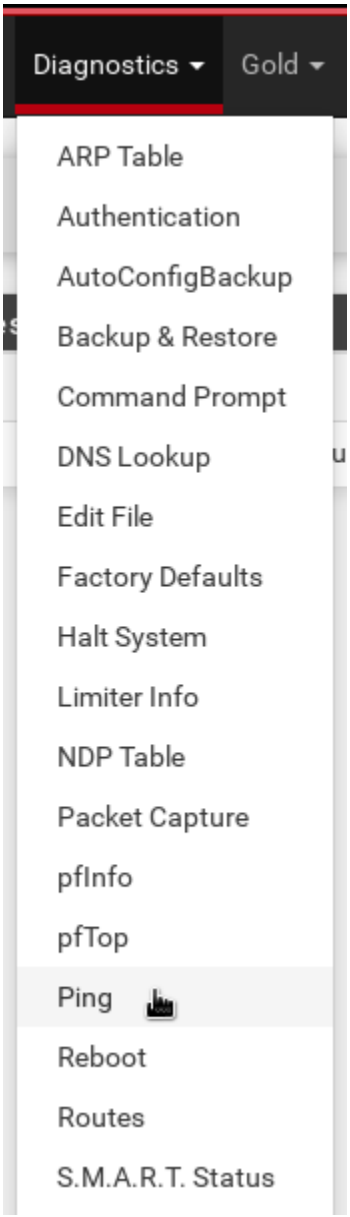
At this point, since you (probably) changed the LAN[1] subnet settings from their defaults, you will no longer be able to connect after reloading the firewall and the next request will probably time out. This is not an error - the firewall has reloaded and is working correctly. To connect to the new LAN[1] interface, unplug and reconnect your network cable to get a new network address assigned via DHCP. Note that if you used a subnet with fewer addresses than `/24`, the default DHCP configuration in pfSense may not work. In this case, you should assign the Admin Workstation a static IP address that is known to be in the subnet to continue.

Now the WebGUI will be available on the Admin Gateway address. Navigate to `https://<Admin Gateway IP>` in the *Unsafe Browser*, and login as before except with the new passphrase you just set for the pfSense WebGUI. Once you've logged in to the WebGUI, you are ready to continue configuring the firewall.

Connect interfaces and test

Now that the initial configuration is completed, you can connect the WAN port without potentially conflicting with the default LAN[1] settings (as explained earlier). Connect the WAN port to the external network. You can watch the WAN entry in the Interfaces table on the pfSense WebGUI homepage to see as it changes from down (red arrow pointing down) to up (green arrow pointing up). This usually takes several seconds. The WAN's IP address will be shown once it comes up.

Finally, test connectivity to make sure you are able to connect to the Internet through the WAN. The easiest way to do this is to use ping (**Diagnostics** → **Ping** in the WebGUI). Enter an external hostname or IP that you expect to be up (e.g. `google.com`) and click "Ping".



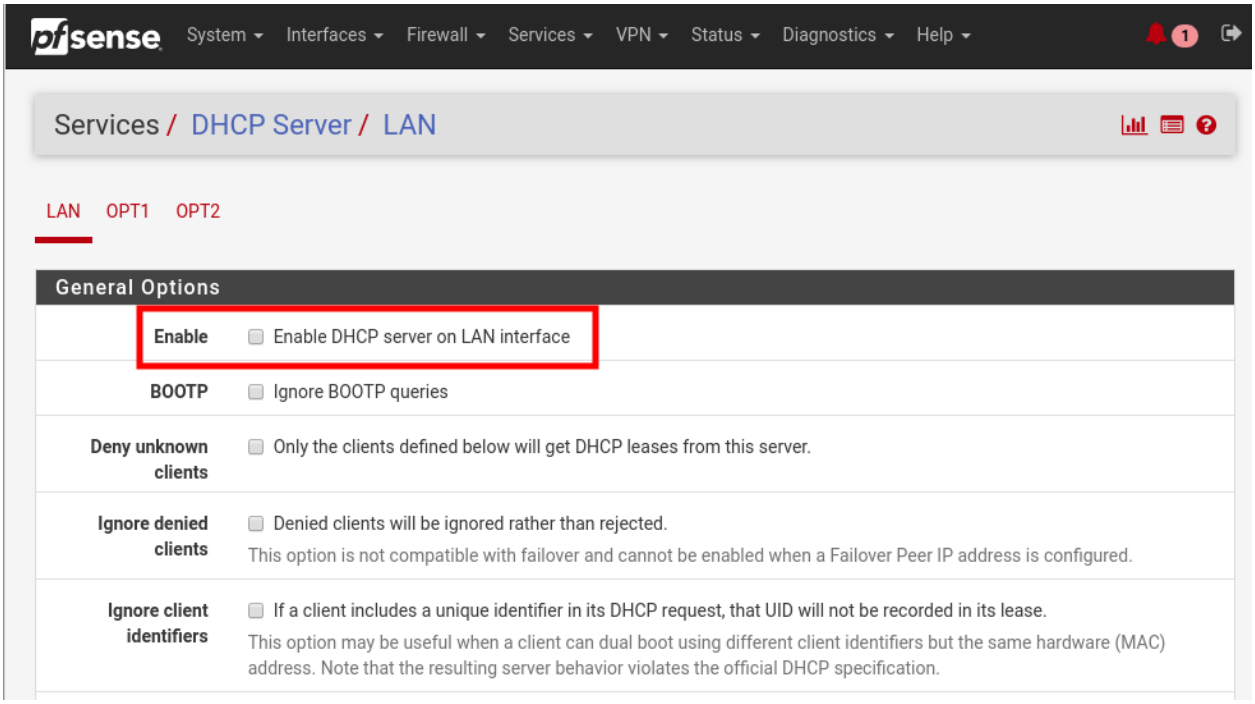
1.24.3 Disable DHCP on the LAN

pfSense runs a DHCP server on the LAN[1] interface by default. At this stage in the documentation, the *Admin Workstation* likely has an IP address assigned via that DHCP server.

In order to tighten the firewall rules as much as possible, we recommend disabling the DHCP server and assigning a static IP address to the Admin Workstation instead.

Disable DHCP server on the firewall

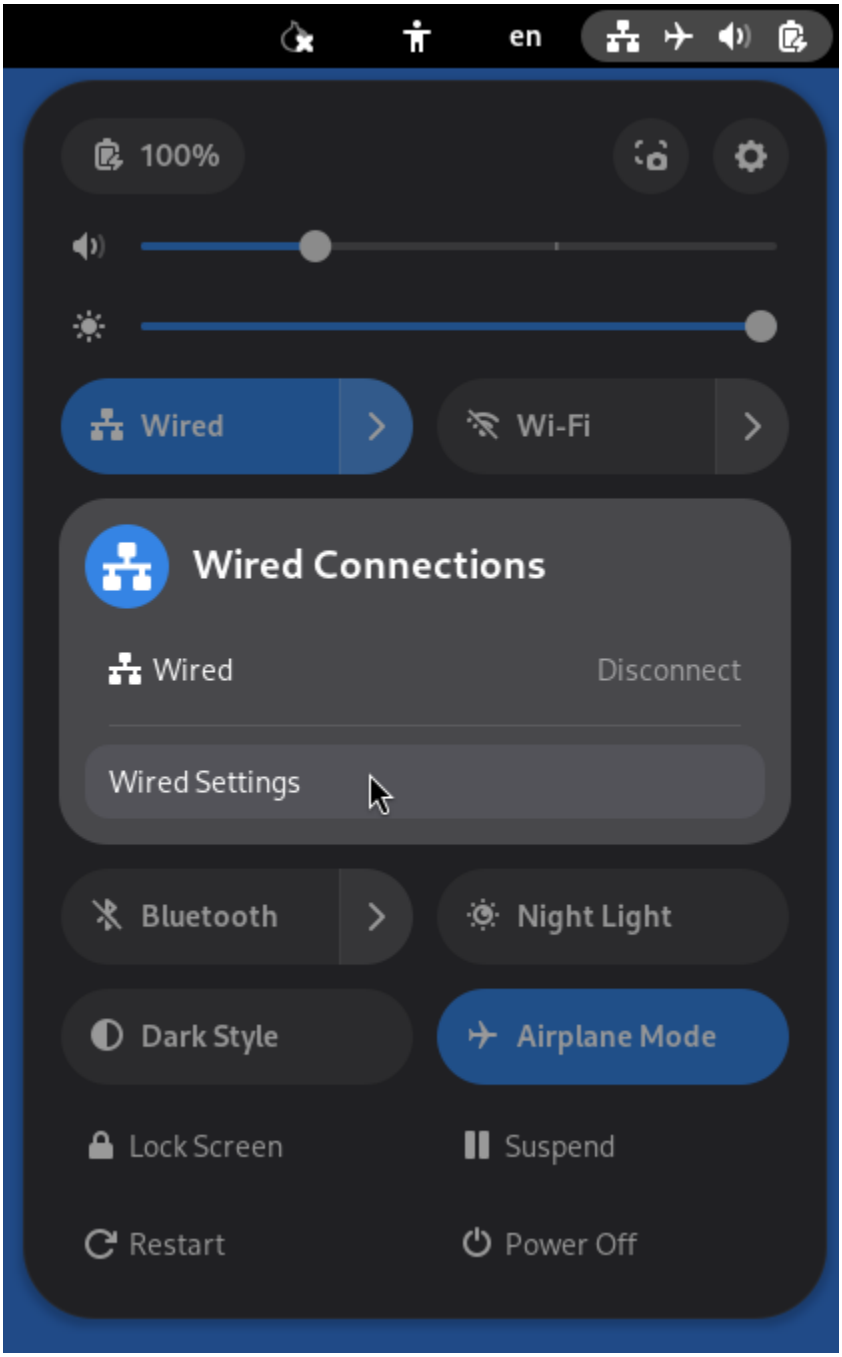
To disable DHCP, navigate to **Services ► DHCP Server** in the pfSense WebGUI. Uncheck the box labeled **Enable DHCP server on LAN interface**, scroll down, and click the **Save** button.



Assign a static IP address to the *Admin Workstation*

Now you will need to assign a static IP to the *Admin Workstation*.

You can easily check your current IP address by *clicking* the top right of the menu bar, clicking on the **Wired Connection** and then clicking **Wired Settings**.



From here you can click on the cog beside the wired network connection:

This will take you to the network settings. Change to the **IPv4** tab. Ensure that **IPv4 Method** is set to **Manual**, and that the **Automatic** switch for **DNS** is in the "off" position, as highlighted in the screenshot below:

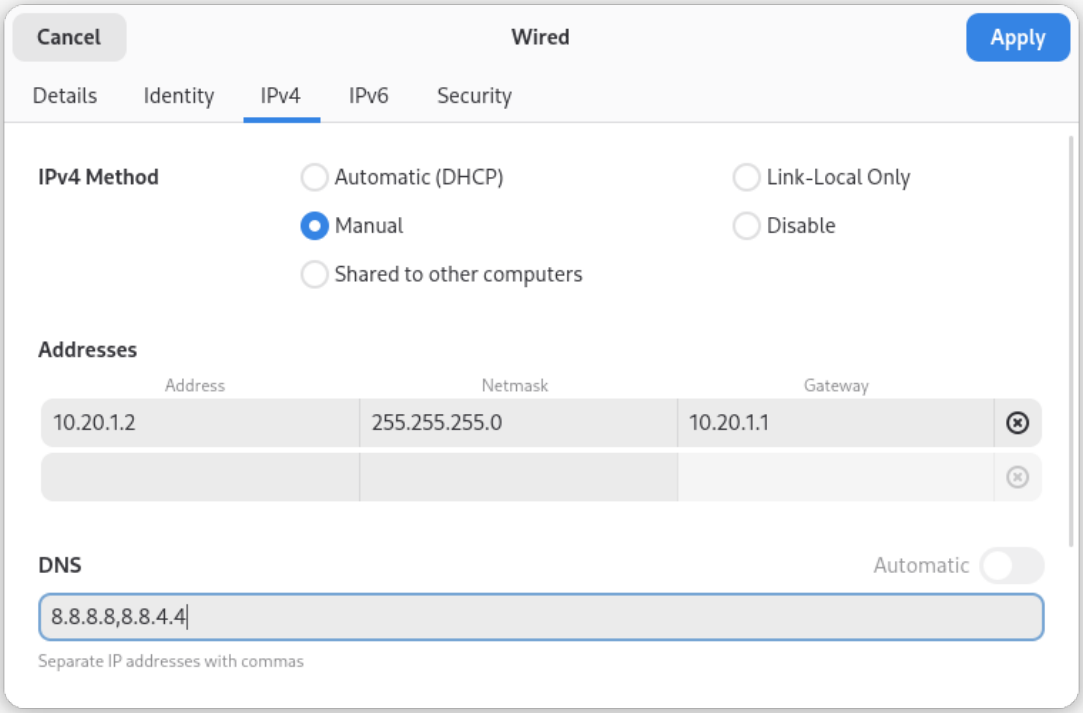
Note

The Unsafe Browser will not launch when using a manual network configuration if it does not have DNS servers configured. This is technically unnecessary for our use case because we are only using it to access IP addresses on the LAN, and do not need to resolve anything with DNS. Nonetheless, you should configure some DNS servers here so you can continue to use the Unsafe Browser to access the WebGUI in future sessions.

We recommend keeping it simple and using the same DNS servers that you used for the network firewall in the setup wizard.

Fill in the static networking information for the *Admin Workstation*:

- Address: `10.20.1.2`
- Netmask: `255.255.255.0`
- Gateway : `10.20.1.1`



Click **Apply**. If the network does not come up within 15 seconds or so, try disconnecting and reconnecting your network cable to trigger the change. You will need you have succeeded in connecting with your new static IP when you are able to connect using the Tor Connection assistant, and you see the message "Connected to Tor successfully".

Troubleshooting: DNS servers and the Unsafe Browser

After saving the new network configuration, you may still encounter the "No DNS servers configured" error when trying to launch the Unsafe Browser. If you encounter this issue, you can resolve it by disconnecting from the network and then reconnecting, which causes the network configuration to be reloaded.

To do this, click the network icon in the system toolbar, and click **Disconnect** under the name of the currently active network connection, which is displayed in bold. After it disconnects, click the network icon again and click the name of the connection to reconnect. You should see a popup notification that says "Connection Established", and the Tor Connection assistant should show the message "Connected to Tor successfully".

For the next step, SecureDrop Configuration, you will manually configure the firewall for SecureDrop, using screenshots as a reference.

1.24.4 SecureDrop configuration

SecureDrop uses the firewall to achieve two primary goals:

1. Isolating SecureDrop from the existing network, which may be compromised (especially if it is a venerable network in a large organization like a newsroom).
2. Isolating the *Application Server* and the *Monitor Server* from each other as much as possible, to reduce attack surface.

In order to use the firewall to isolate the *Application Server* and the *Monitor Server* from each other, we need to connect them to separate interfaces, and then set up firewall rules that allow them to communicate.

Set up the firewall rules

Since there are a variety of firewalls with different configuration interfaces and underlying sets of software, we cannot provide a set of network firewall rules to match every use case.

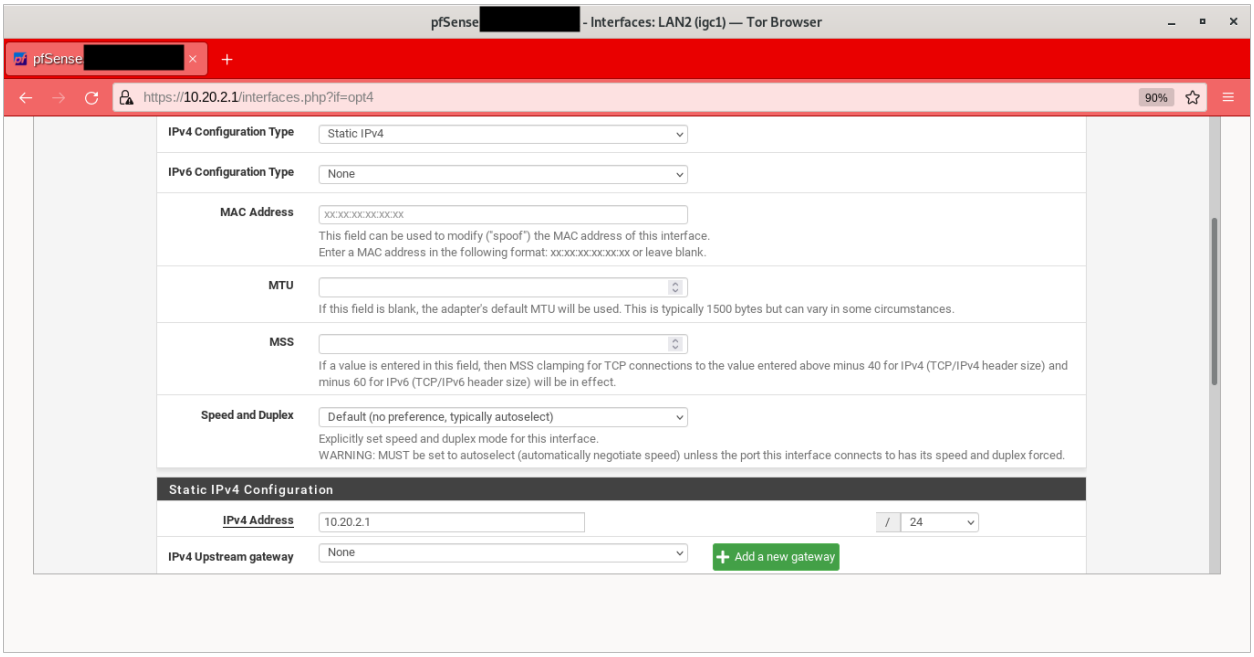
The easiest way to set up your firewall rules is to look at the screenshots of a correctly configured firewall and edit the interfaces, aliases, and firewall rules on your firewall to match them.

Set up LAN2

We set up the LAN[1] interface during the initial configuration. We now need to set up the LAN2 interface for the *Application Server*. Start by connecting the *Application Server* to the LAN2 port. Then use the WebGUI to configure the LAN2 interface. Go to **Interfaces ► LAN2**, and check the box to **Enable Interface**. Use these settings:

- IPv4 Configuration Type: Static IPv4
- IPv4 Address: `10.20.2.1` (Application Gateway IP)

Make sure that the CIDR routing prefix is correct (`/24`). Leave everything else as the default. **Save** and **Apply Changes**.

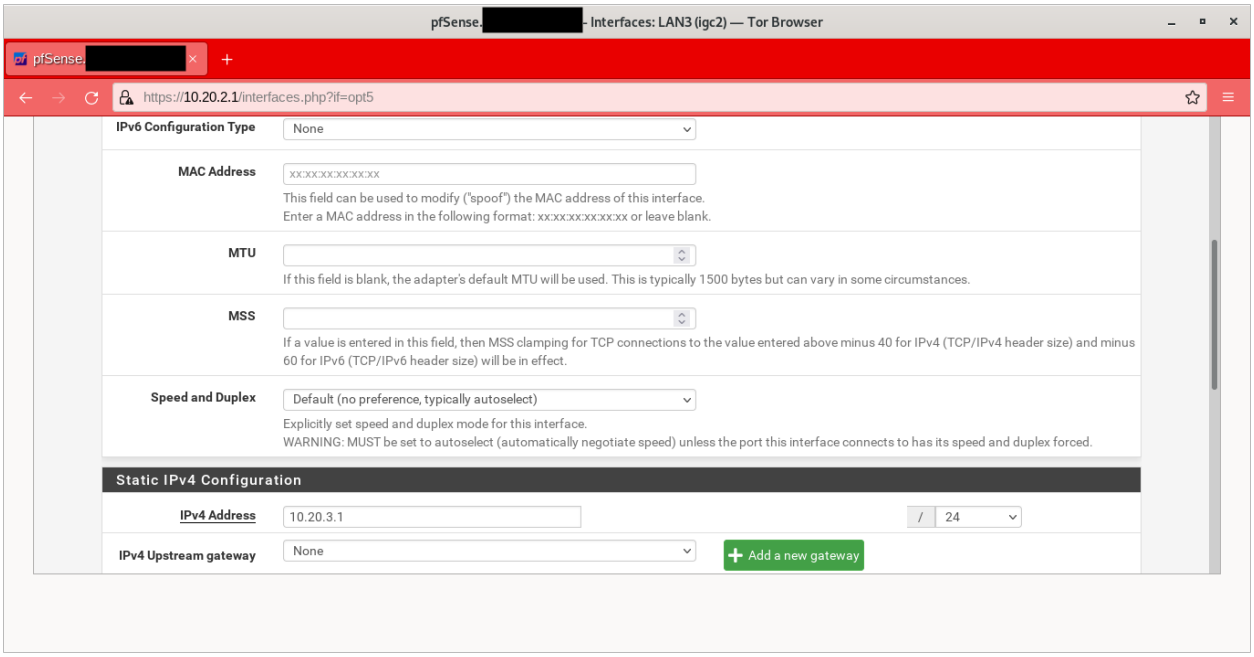


Set up LAN3

Next, you will have to enable the LAN3 interface. Go to **Interfaces ▶ LAN3**, and check the box to **Enable Interface**. LAN3 interface is set up similarly to how we set up LAN2 in the previous section. Use these settings:

- IPv4 Configuration Type: Static IPv4
- IPv4 Address: `10.20.3.1` (Monitor Gateway IP)

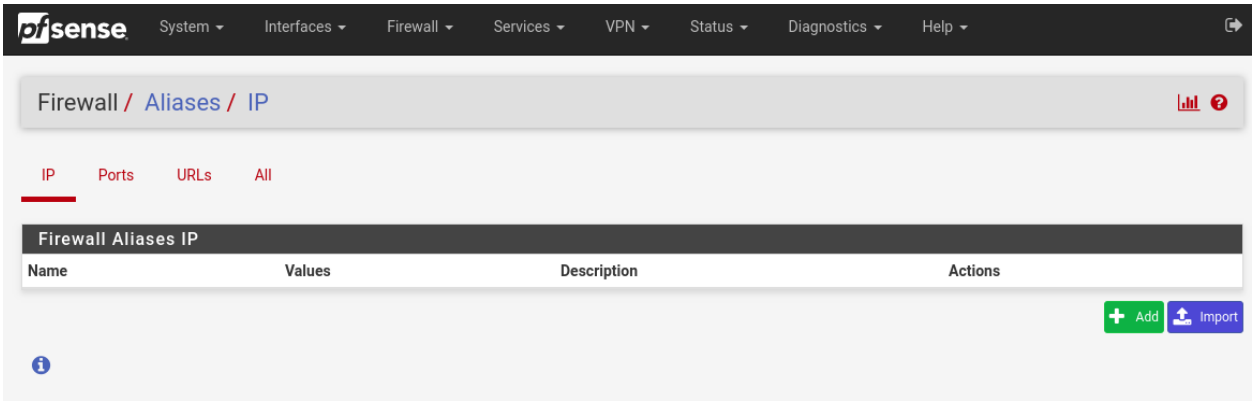
Make sure that the CIDR routing prefix is correct (`/24`). Leave everything else as the default. **Save** and **Apply Changes**.



Use screenshots of firewall configuration

Here are some example screenshots of a working pfSense firewall configuration. You will add the firewall rules until they match what is shown on the screenshots.

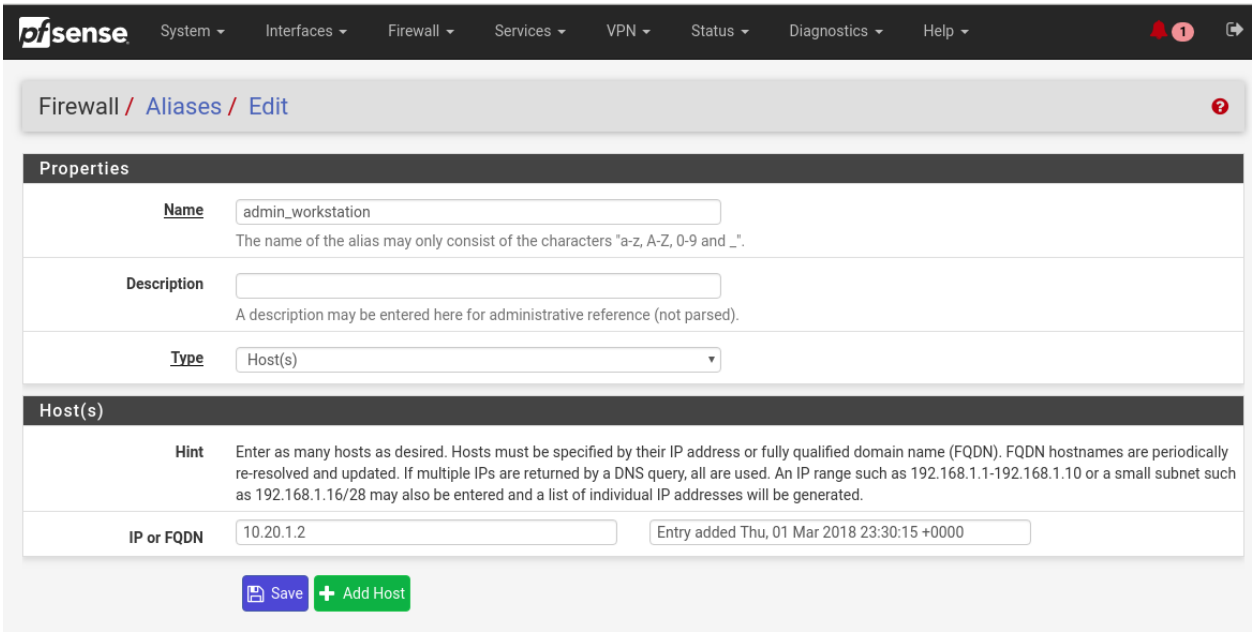
First, we will configure IP and port aliases. Navigate to **Firewall ▶ Aliases** and you should see a screen with no currently defined IP aliases:



Next you will click **Add** to add each IP alias. You should leave the **Type** as **Host**. Make aliases for the following:

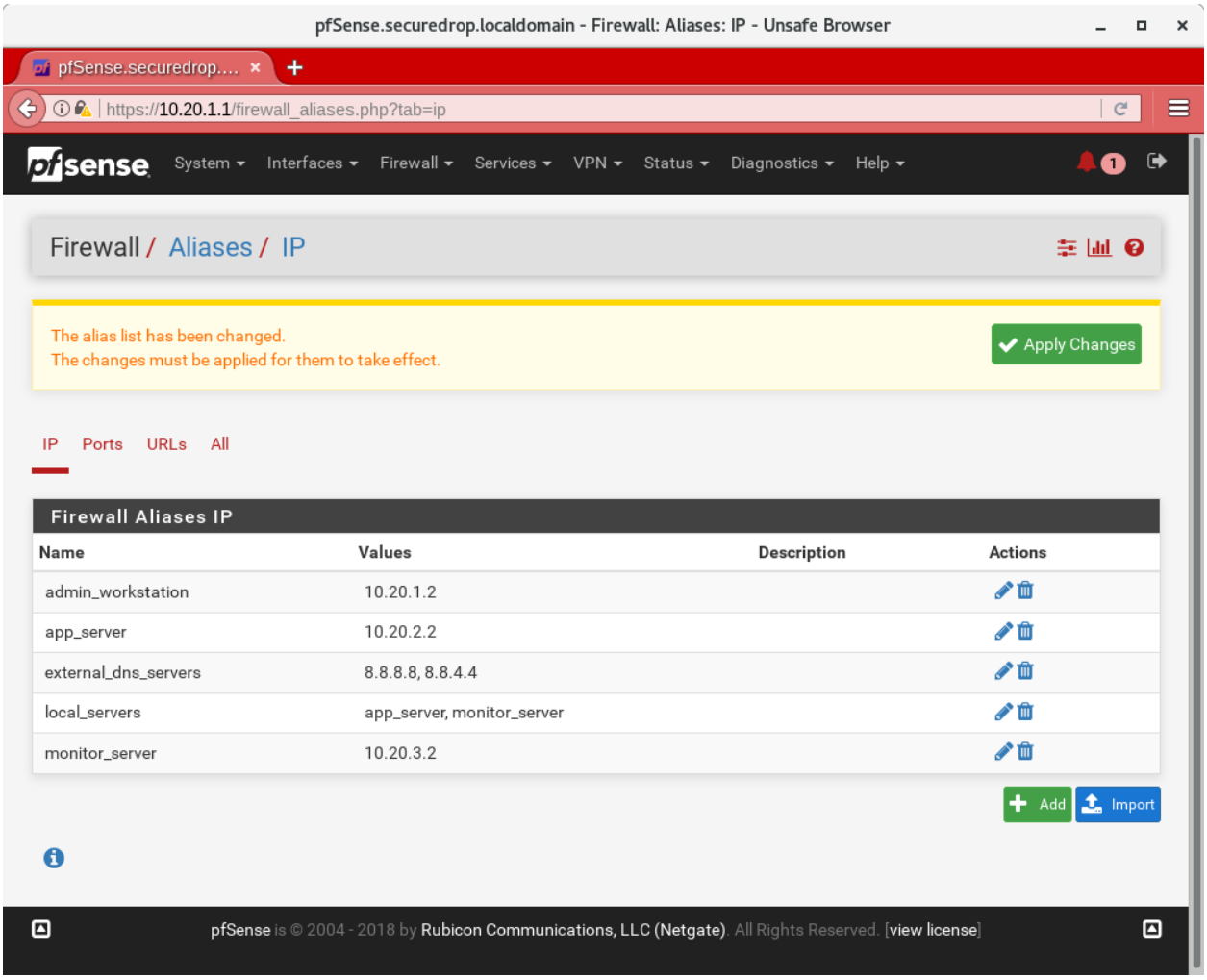
- `admin_workstation`: `10.20.1.2`

- `app_server`: `10.20.2.2`
- `external_dns_servers`: `8.8.8.8, 8.8.4.4`
- `monitor_server`: `10.20.3.2`
- `local_servers`: `app_server, monitor_server`

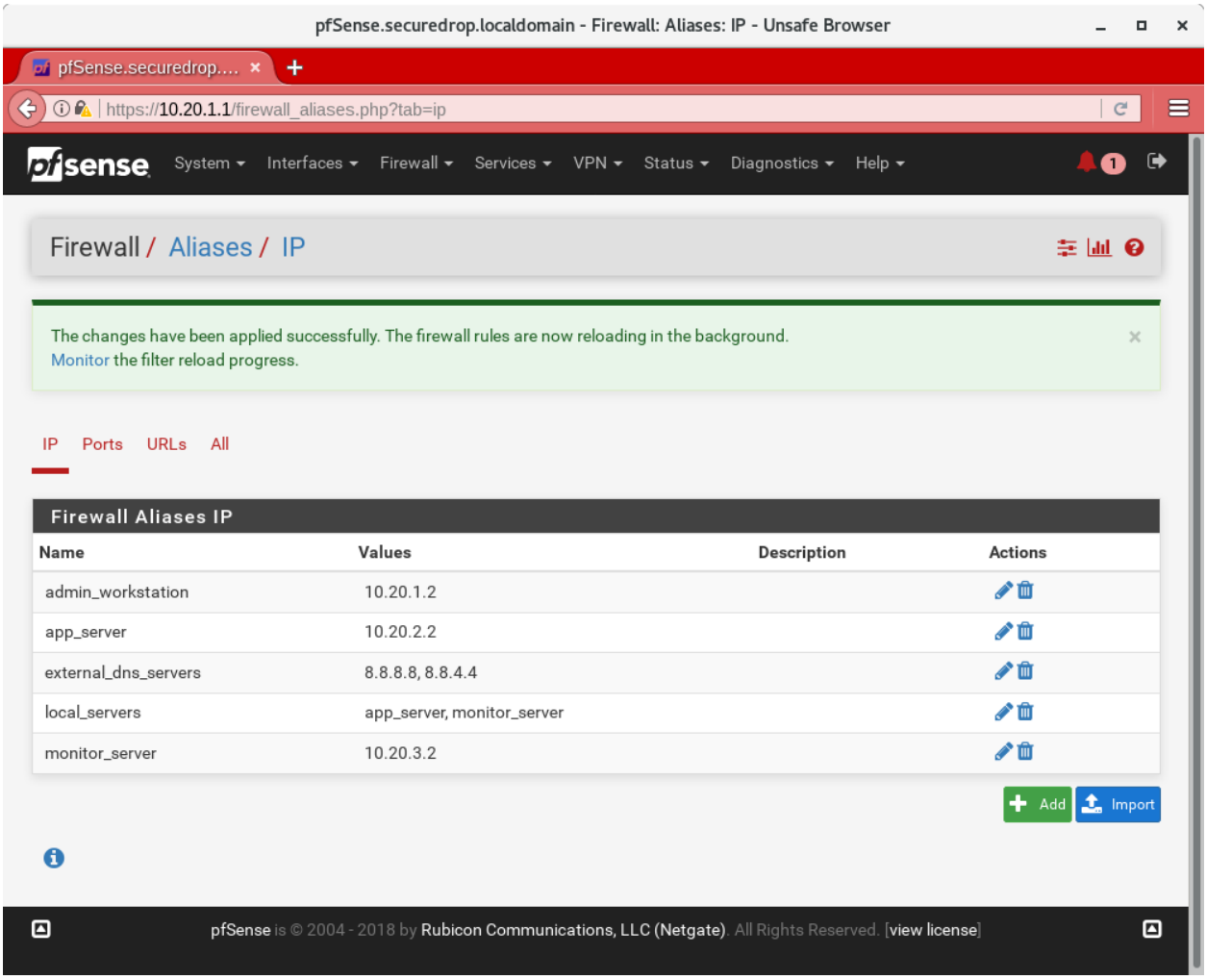


Click **Save** to add the alias.

Keep adding aliases until the screenshot matches what is shown here:



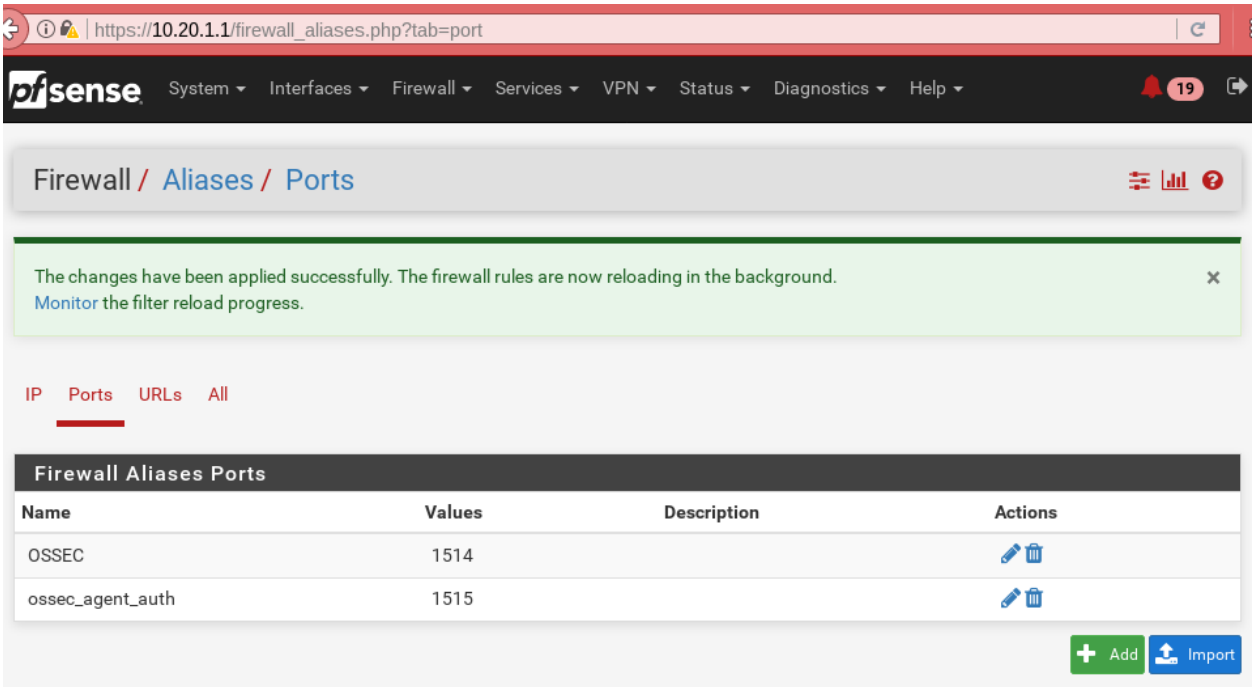
Finally, click **Apply Changes**. This will save your changes. You should see a message "The changes have been applied successfully":



Next click “Ports” for the port aliases, and add the following ports:

- OSSEC: `1514`
- ossec_agent_auth: `1515`

Your configuration should match this screenshot:



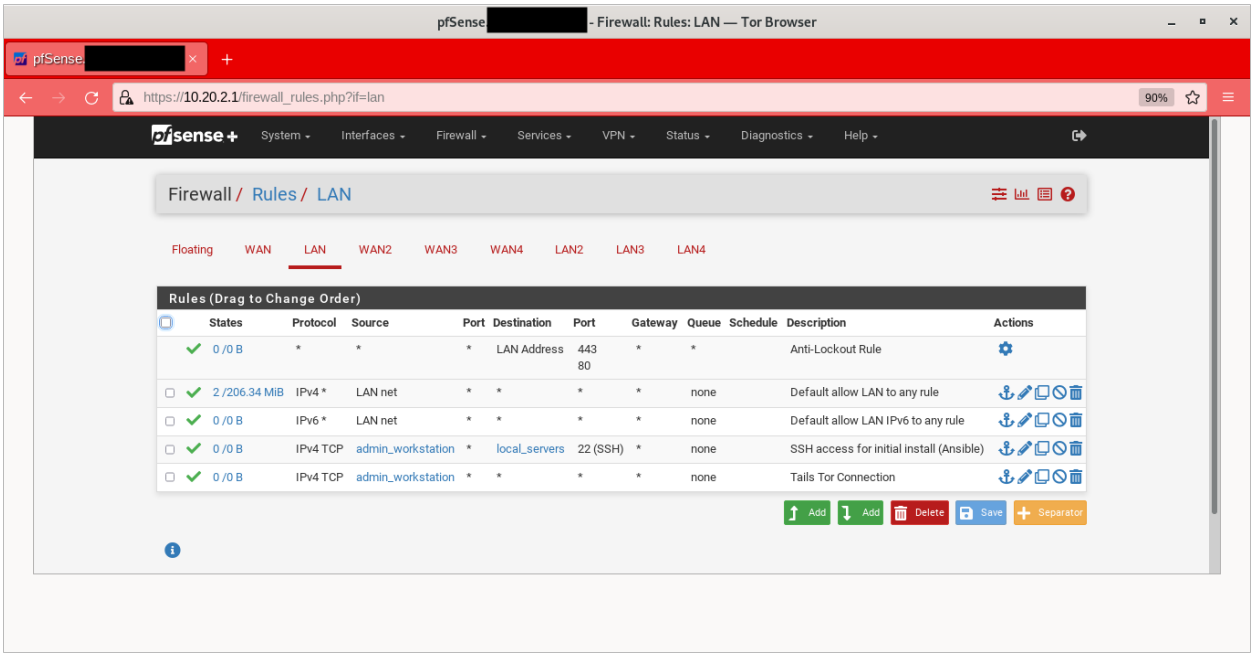
Next we will configure firewall rules for each interface. Navigate to **Firewall ► Rules** to add firewall rules for the LAN1, LAN2, and LAN3 interfaces.

Warning

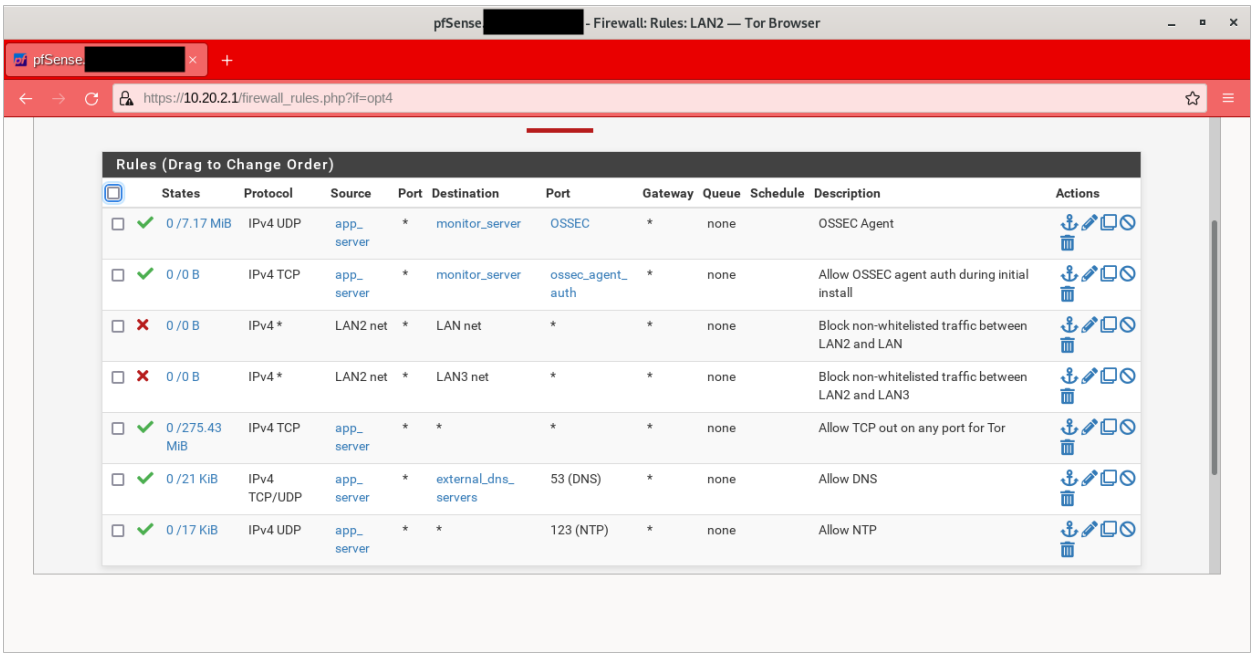
Be sure not to delete the Anti-Lockout Rule on the LAN1 interface. Deleting this rule will lock you out of the pfSense WebGUI.

Add or remove rules until they match the following screenshots by clicking **Add** to add a rule.

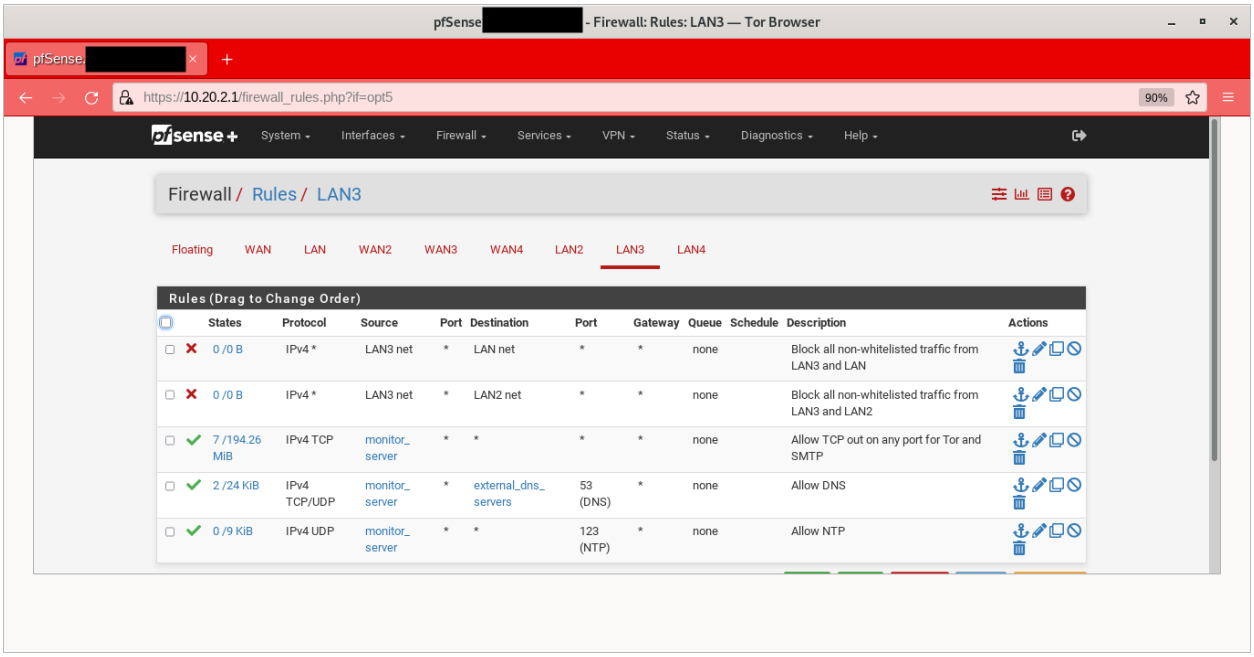
LAN[1] interface:



LAN2 interface:



LAN3 interface:



Finally, click **Apply Changes**. This will save your changes. You should see a message "The changes have been applied successfully". Once you've set up the firewall, exit the Unsafe Browser, and continue with the "Keeping pfSense up to date" section below.

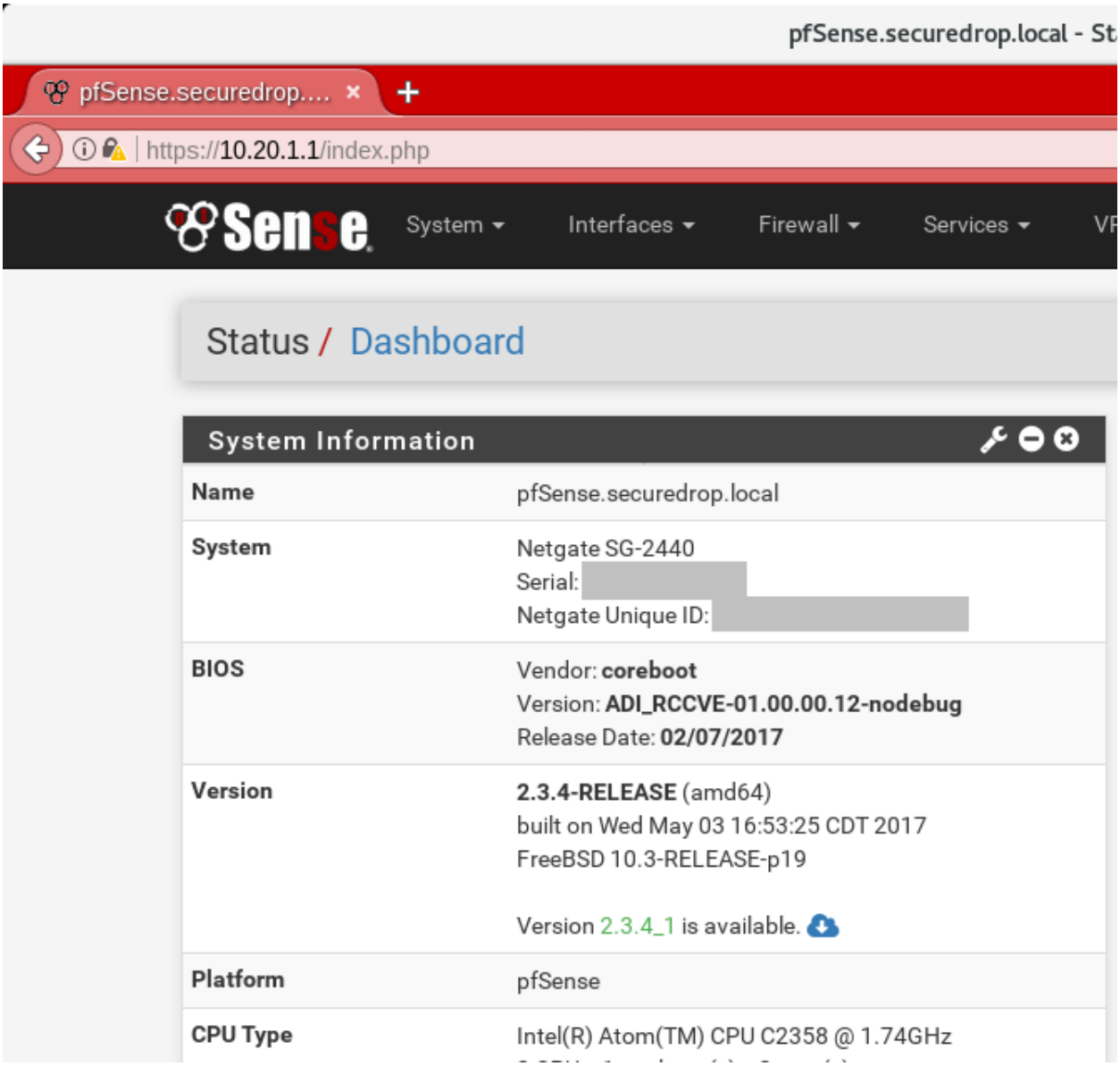
1.24.5 Tips for setting up pfSense firewall rules

Here are some general tips for setting up pfSense firewall rules:

1. Create aliases for the repeated values (IPs and ports).
2. pfSense is a stateful firewall, which means that you don't need corresponding rules to allow incoming traffic in response to outgoing traffic (like you would in, e.g. iptables with `--state ESTABLISHED,RELATED`). pfSense does this for you automatically.
3. You should create the rules *on the interface where the traffic originates*.
4. Make sure you delete the default "allow all" rule on the LAN interface. Leave the "Anti-Lockout" rule enabled.
5. Any traffic that is not explicitly passed is logged and dropped by default in pfSense, so you don't need to add explicit rules (iptables `LOGNDROP`) for that.
6. Since some of the rules are almost identical except for whether they allow traffic from the *Application Server* or the *Monitor Server*, you can use the "add a new rule based on this one" button to save time creating a copy of the rule on the other interface.
7. If you are troubleshooting connectivity, the firewall logs can be very helpful. You can find them in the WebGUI in *Status* → *System Logs* → *Firewall*.

1.24.6 Keeping pfSense up to date

Periodically, the pfSense project maintainers release an update to the pfSense software running on your firewall. You will be notified by the appearance of text saying that there is a new version in the **Version** section of the "Status: Dashboard" page (the home page of the WebGUI).

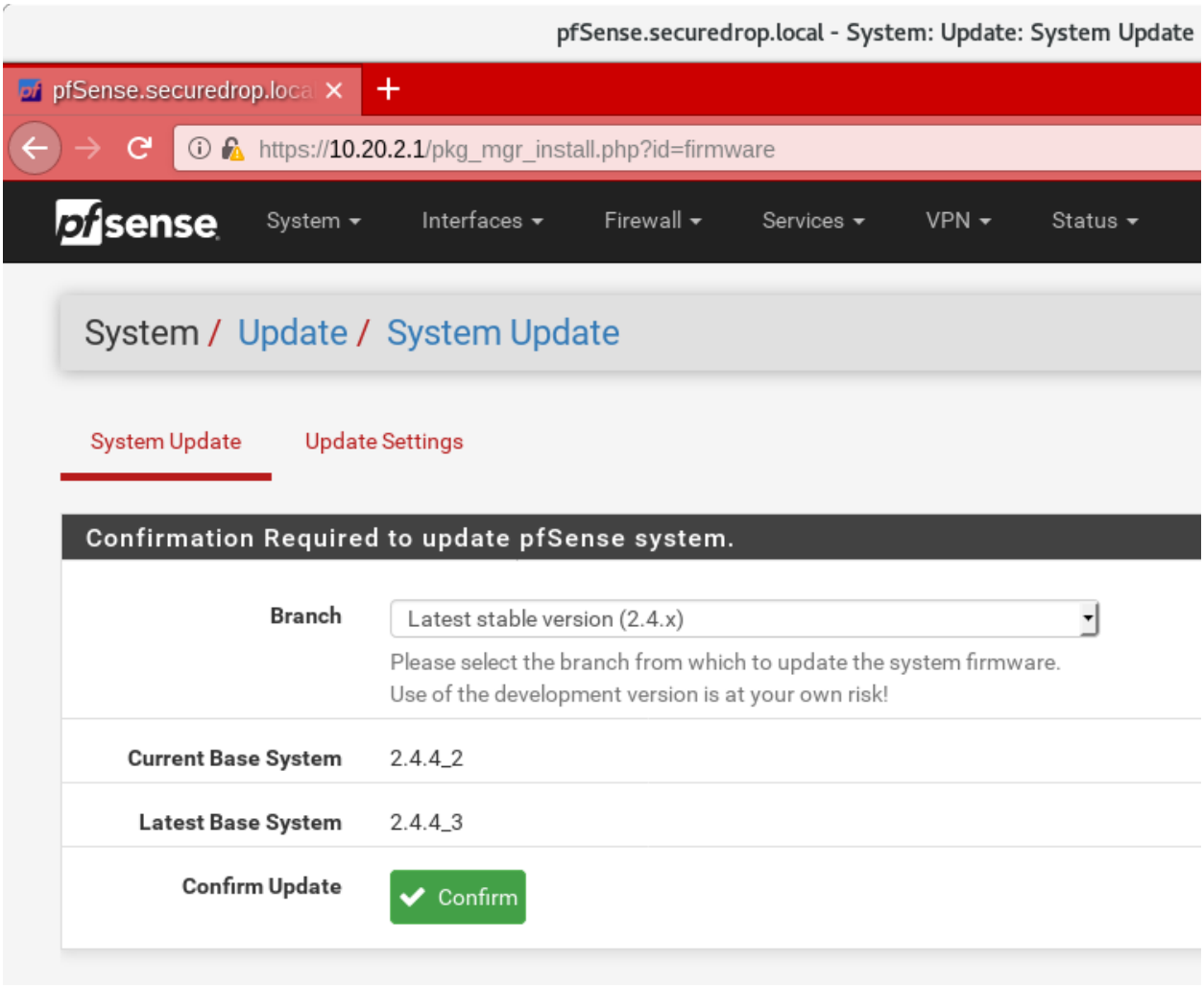


If you see that an update is available, we recommend installing it. Most of these updates are for minor bugfixes, but occasionally they can contain important security fixes. You should keep apprised of updates yourself by checking the pfSense Blog posts with the "releases" tag.

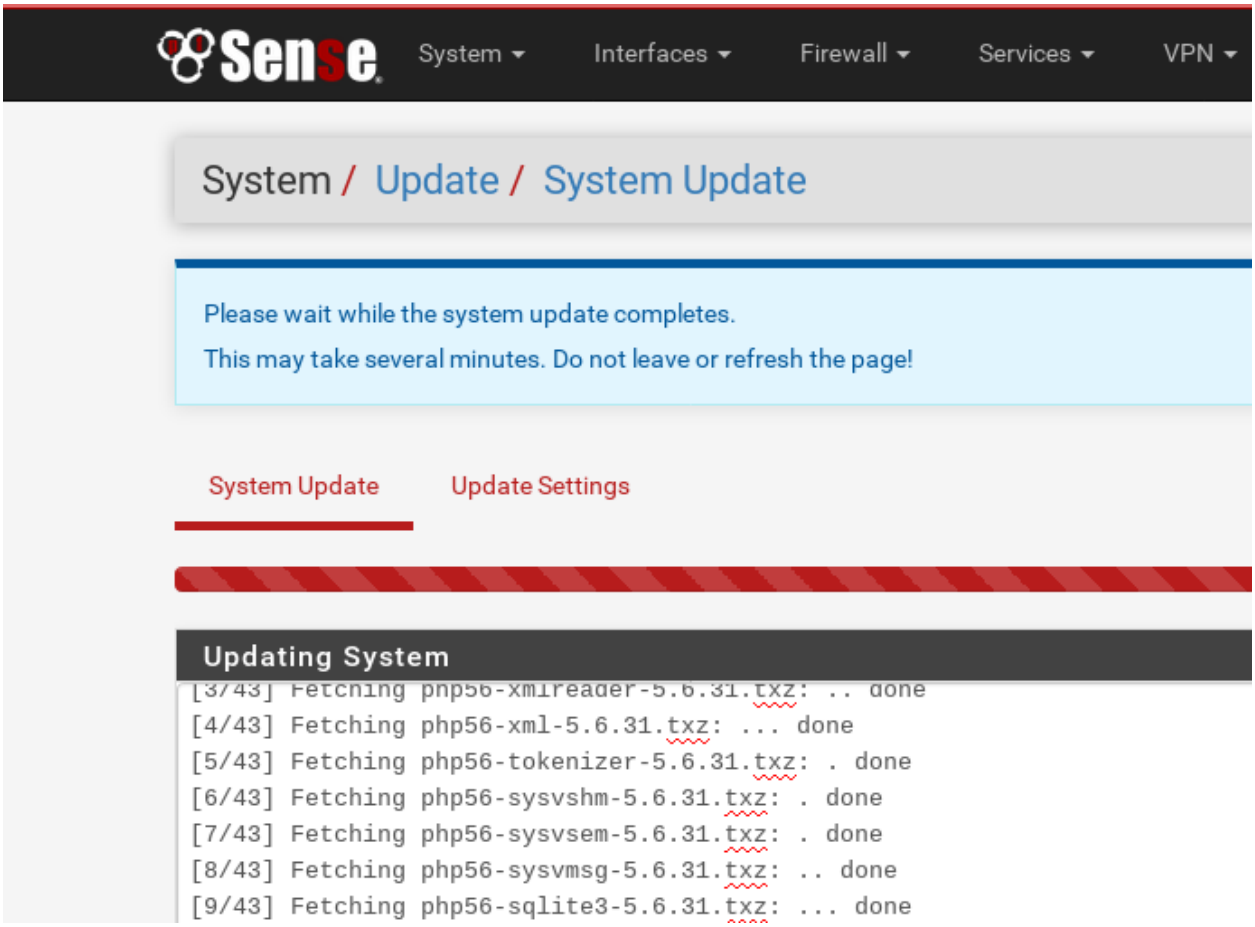
Note

You can subscribe to email updates on https://www.netgate.com.

To install the update, click the Download icon next to the update then click the "Confirm" button:



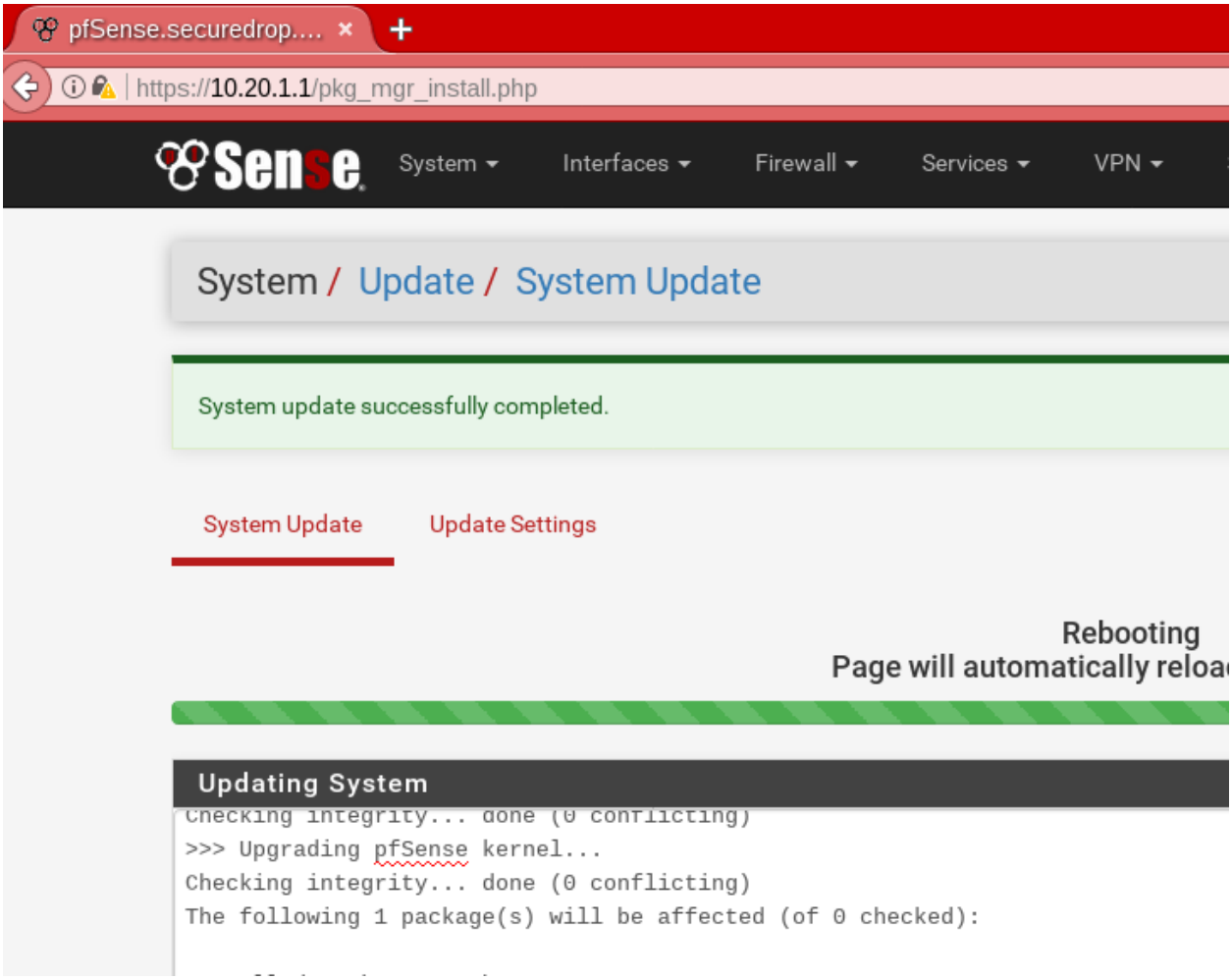
You will see a page with a progress bar while pfSense performs the upgrade:



Note

This may take a while, so be patient!

Once it is complete, you will see a notification of successful upgrade:



The *Network Firewall* configuration is now complete, allowing you to move to the next step: *setting up the servers.*

1.25 Setting up an OPNSense network firewall

1.25.1 Before you begin

First, consider how the firewall will be connected to the Internet. You will need to provision several unique subnets, which should not conflict with the network configuration on the WAN interface. If you are unsure, consult your local system administrator.

Many firewalls, including the recommended OPNSense device, automatically set up the LAN interface on `192.168.1.1/24`. This particular private network is also a very common choice for home and office routers. If you are connecting the firewall to a router with the same subnet (common in a small office, home, or testing environment), you will probably be unable to connect to the network at first. However, you will be able to connect from the LAN to the firewall's Web GUI, and from there you will be able to configure the network so it is working correctly.

The recommended TekLager APU4D4 has 4 NICs: WAN, LAN, OPT1, and OPT2. This allows for a dedicated port on the network firewall for each component of SecureDrop (*Application Server*, *Monitor Server*, and *Admin Workstation*).

Depending on your network configuration, you should define the following values before continuing.

- Admin Subnet: `10.20.1.0/24`
- Admin Gateway: `10.20.1.1`

- Admin Workstation: `10.20.1.2`
- Application Subnet: `10.20.2.0/24`
- Application Gateway: `10.20.2.1`
- Application Server (OPT1): `10.20.2.2`
- Monitor Subnet: `10.20.3.0/24`
- Monitor Gateway: `10.20.3.1`
- Monitor Server (OPT2) : `10.20.3.2`

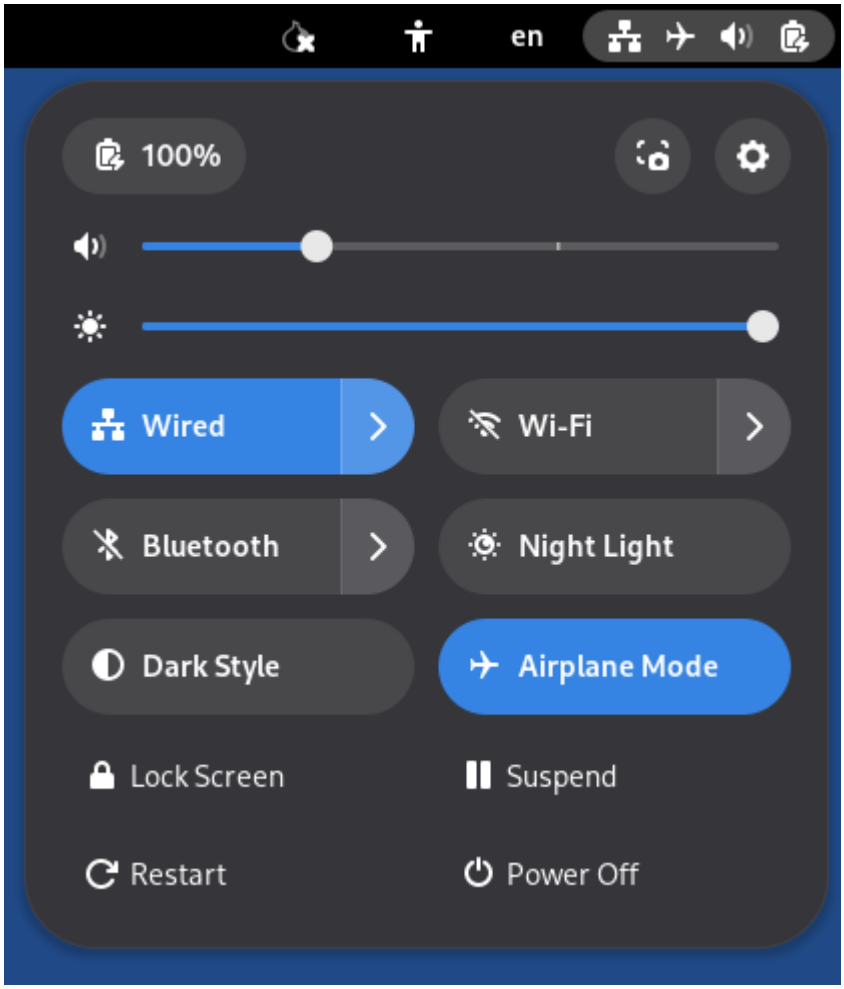
1.25.2 Initial configuration

Unpack the firewall, connect the power, and power on the device.

We will use the OPNSense Web GUI to do the initial configuration of the network firewall.

Connect to the OPNSense web GUI

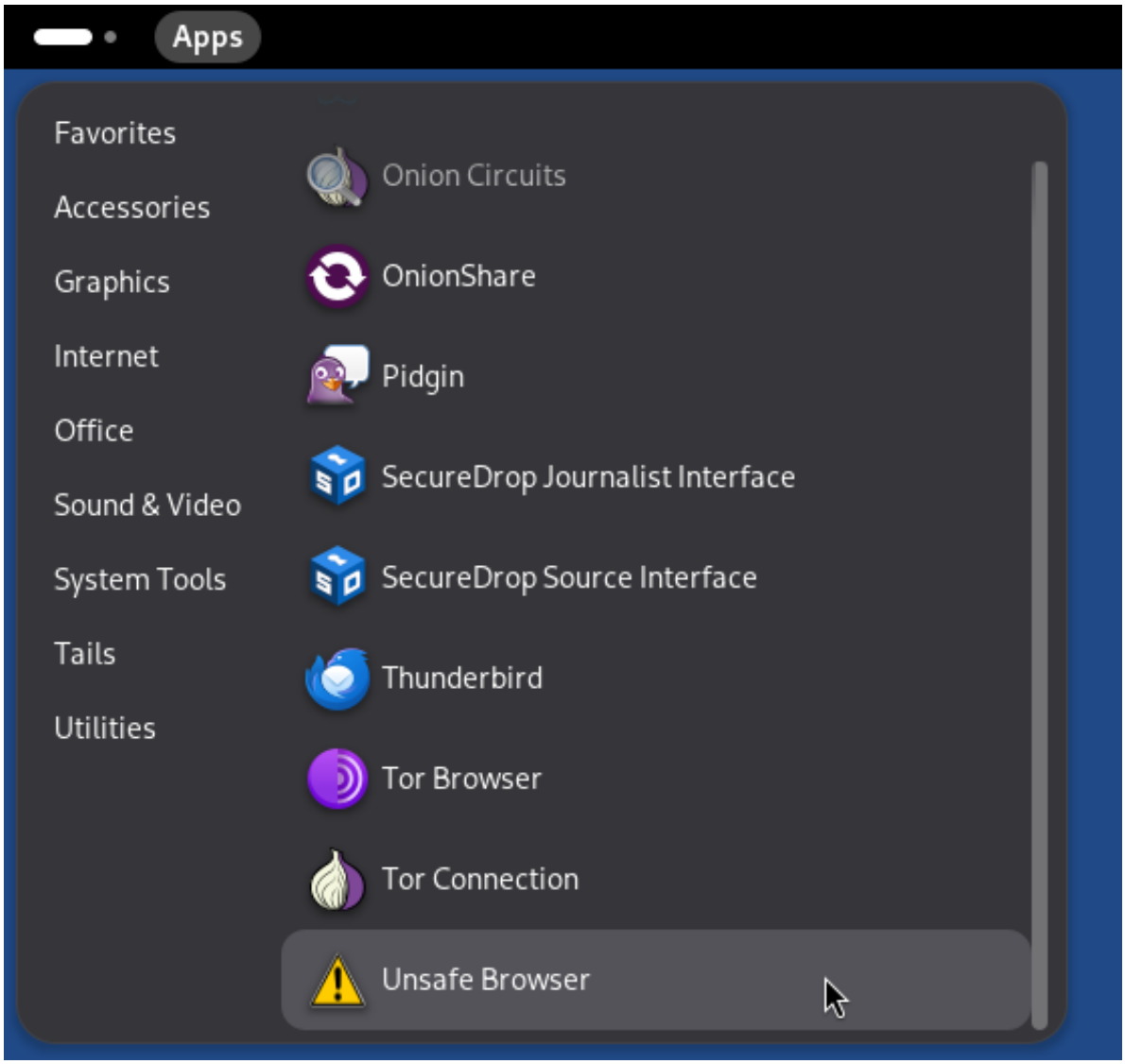
1. If you have not already done so, boot the *Admin Workstation*.
2. Connect the *Admin Workstation* to the LAN interface. You should see a popup notification in Tails that says "Connection Established". If you click on the network icon in the upper right of the Tails Desktop, you should see that the "Wired Connection" is active:



Warning

Make sure your *only* active connection is the one you just established with the network firewall. If you are connected to another network at the same time (e.g. a wireless network), you may encounter problems trying to connect the firewall's Web GUI.

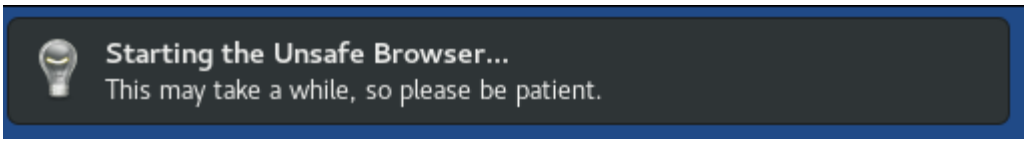
3. Launch the **Unsafe Browser** from the menu bar: **Apps ► Internet ► Unsafe Browser**.



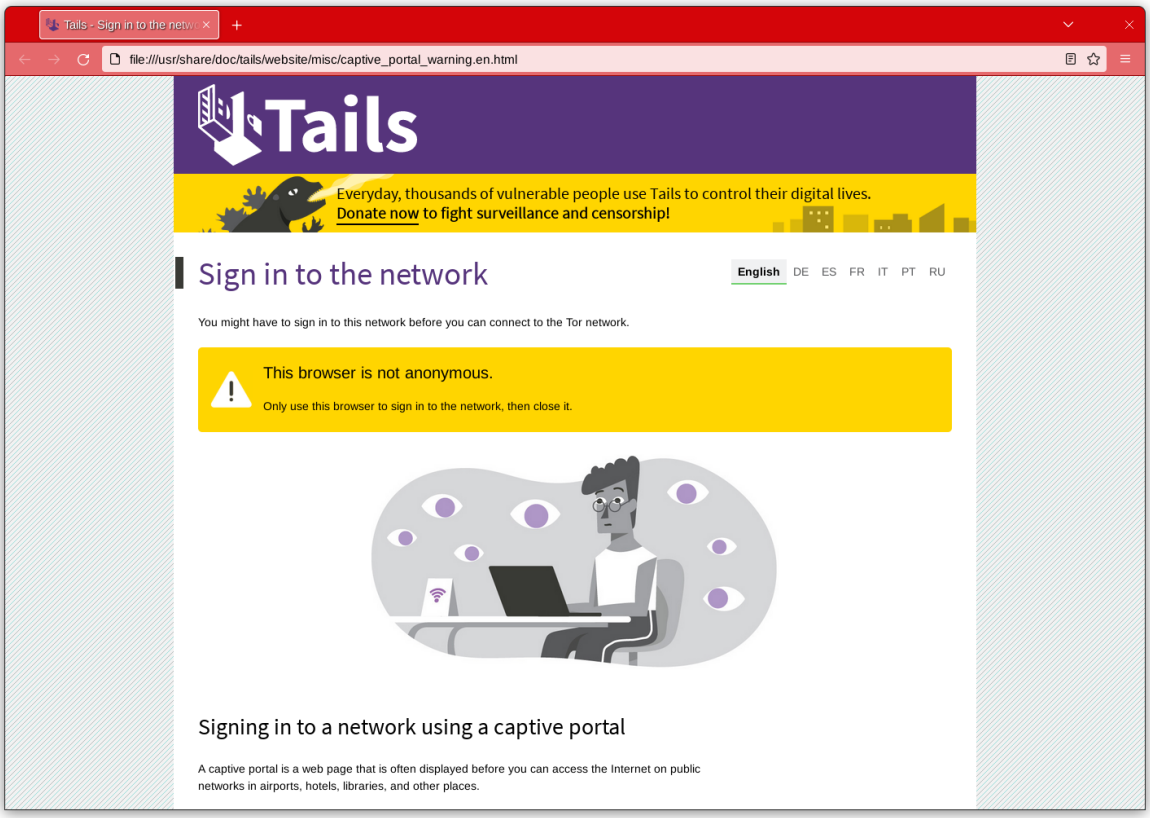
Note

The *Unsafe Browser* is, as the name suggests, **unsafe** (its traffic is not routed through Tor). However, it is the only option because Tails intentionally disables LAN access in the **Tor Browser**.

4. You will see a pop-up notification that says "Starting the Unsafe Browser…"



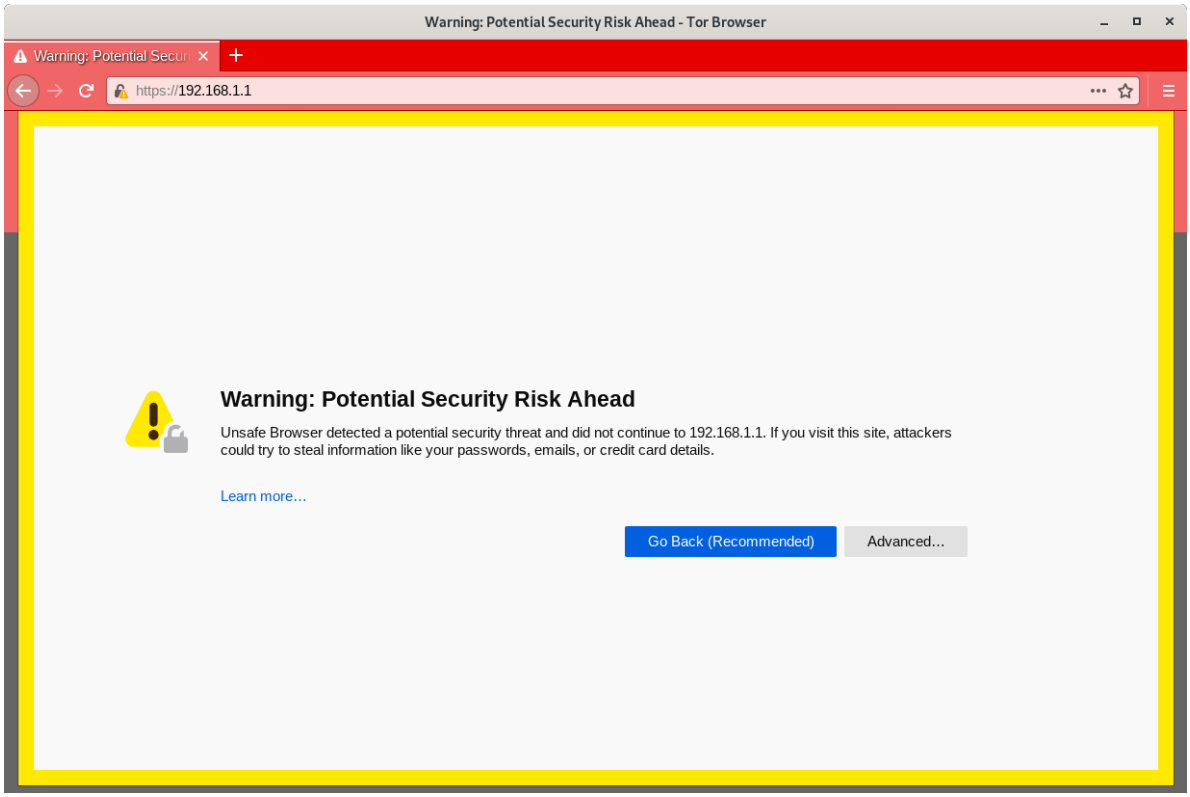
5. After a few seconds, the Unsafe Browser should launch. The window has a bright red border to remind you to be careful when using it. You should close it once you're done configuring the firewall and use Tor Browser for any other web browsing you might do on the *Admin Workstation*.



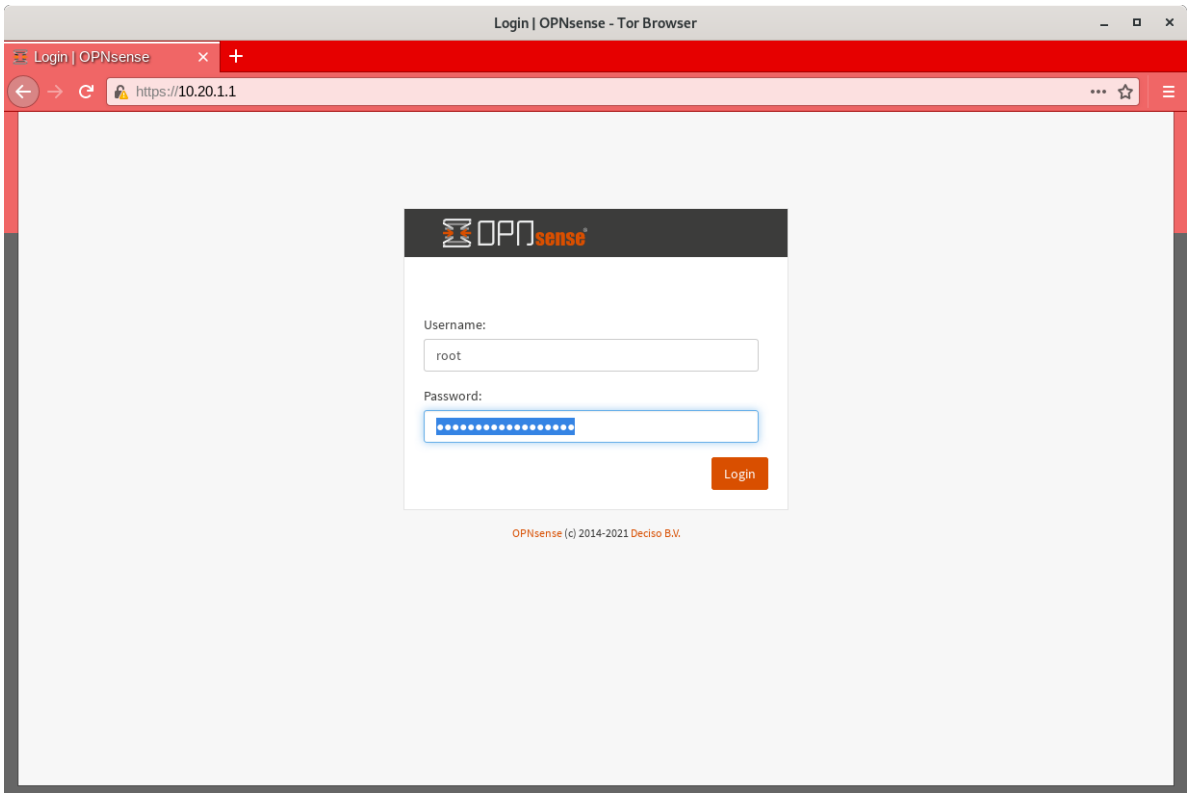
6. Navigate to the OPNSense Web GUI in the *Unsafe Browser*: `https://192.168.1.1`

> **Note**
>
> If you have trouble connecting, go to your network settings and make sure that you have an IPv4 address in the `192.168.1.1/24` range. You may need to turn on DHCP, else you can manually configure a static IPv4 address of `192.168.1.x` with a subnet mask of `255.255.255.0`. However, make sure not to configure your Tails device to have the same IP as the firewall (`192.168.1.1`).

7. The firewall uses a self-signed certificate, so you will see a "This Connection Is Untrusted" warning when you connect. This is expected. You can safely continue by clicking **Advanced** and **Accept the Risk and Continue**.



8. You should see the login page for the OPNSense GUI. Log in with the default username and passphrase (`root` / `opnsense`).



If this is your first time logging in to the firewall, the setup wizard will be displayed. You should not step through it at

this point, however, as there are other tasks to complete. To exit, click the OPNSense logo in the top left corner of the screen.

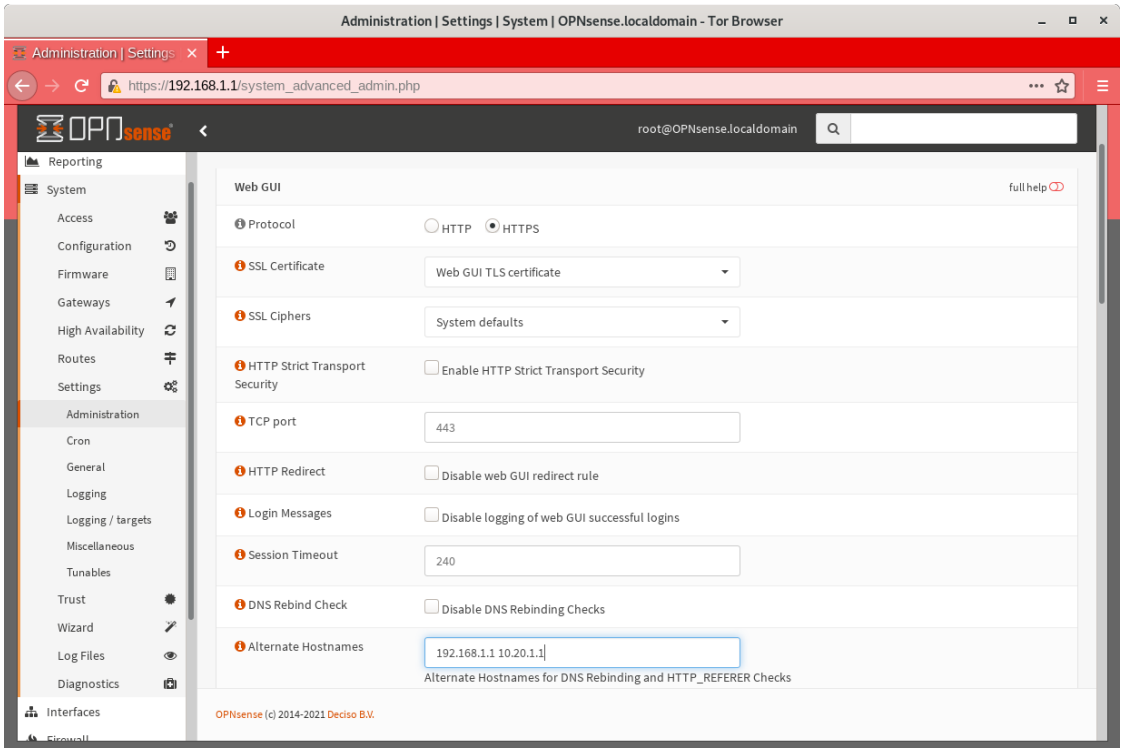
Set a strong password

Navigate to **System ▶ Access ▶ Users** and click the edit button for the `root` user. On the subsequent page, set a strong admin password. We recommend generating a strong passphrase with KeePassXC and saving it in the Tails Persistent folder using the provided KeePassXC database template. *Two-Factor Authentication* will be enabled in a later step.

Set alternate hostnames

Before you can set up the hardware firewall, you will need to set the **Alternate Hostnames** setting.

First, navigate to **System ▶ Settings ▶ Administration**. In the **Web GUI** section, update the **Alternate Hostnames** field with the values `192.168.1.1` and the IP address of the *Admin Gateway* (`10.20.1.1` if you are using the recommended default values), separated by a space.



Finally, scroll to the bottom of the page and click **Save**.

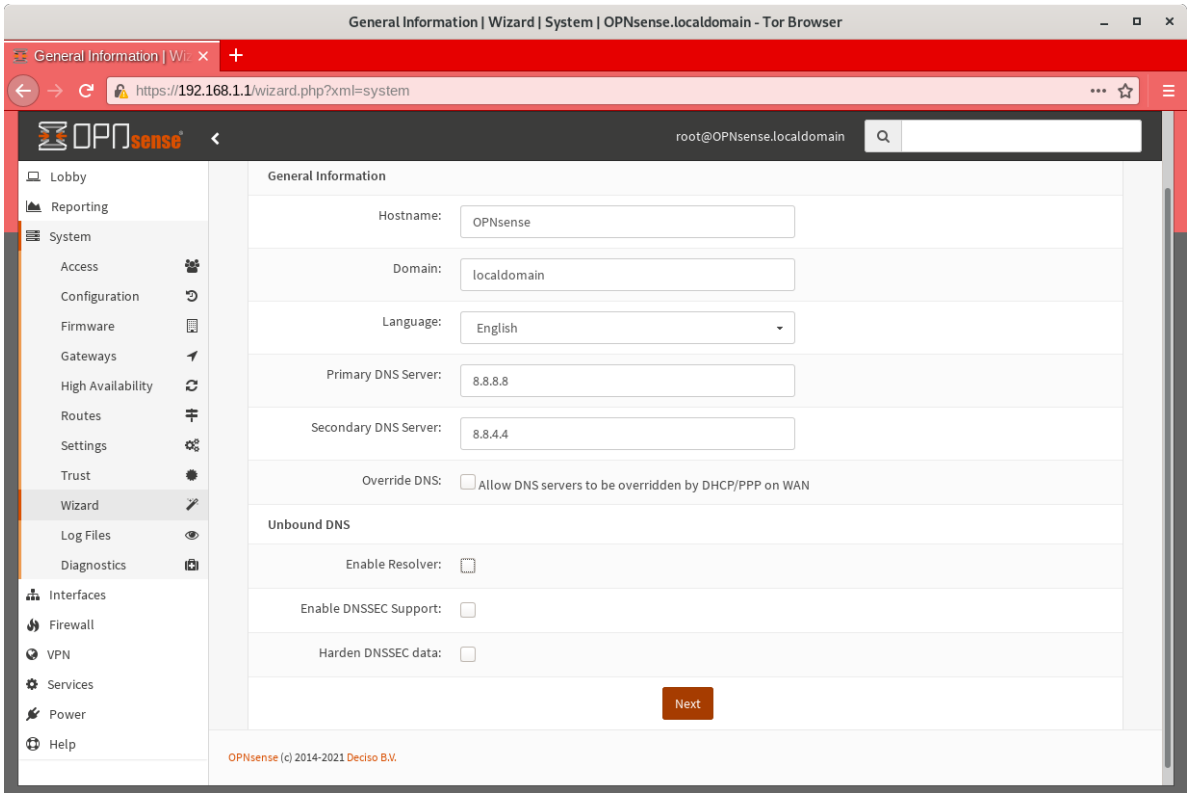
Configure interfaces via the Setup Wizard

To start the OPNSense Setup Wizard, navigate to **System ▶ Wizard** and click **Next**.

1. **General Information**: Leave your hostname as the default, `OPNsense`. There is no relevant domain for SecureDrop, so we recommend setting this to `securedrop.local` or something similar. Use your preferred DNS servers. If you don't know what DNS servers to use, we recommend using Google's DNS servers: `8.8.8.8` and `8.8.4.4`. Uncheck the **Override DNS** checkbox.

   In the **Unbound DNS** section, uncheck **Enable Resolver**.

   Click **Next**.

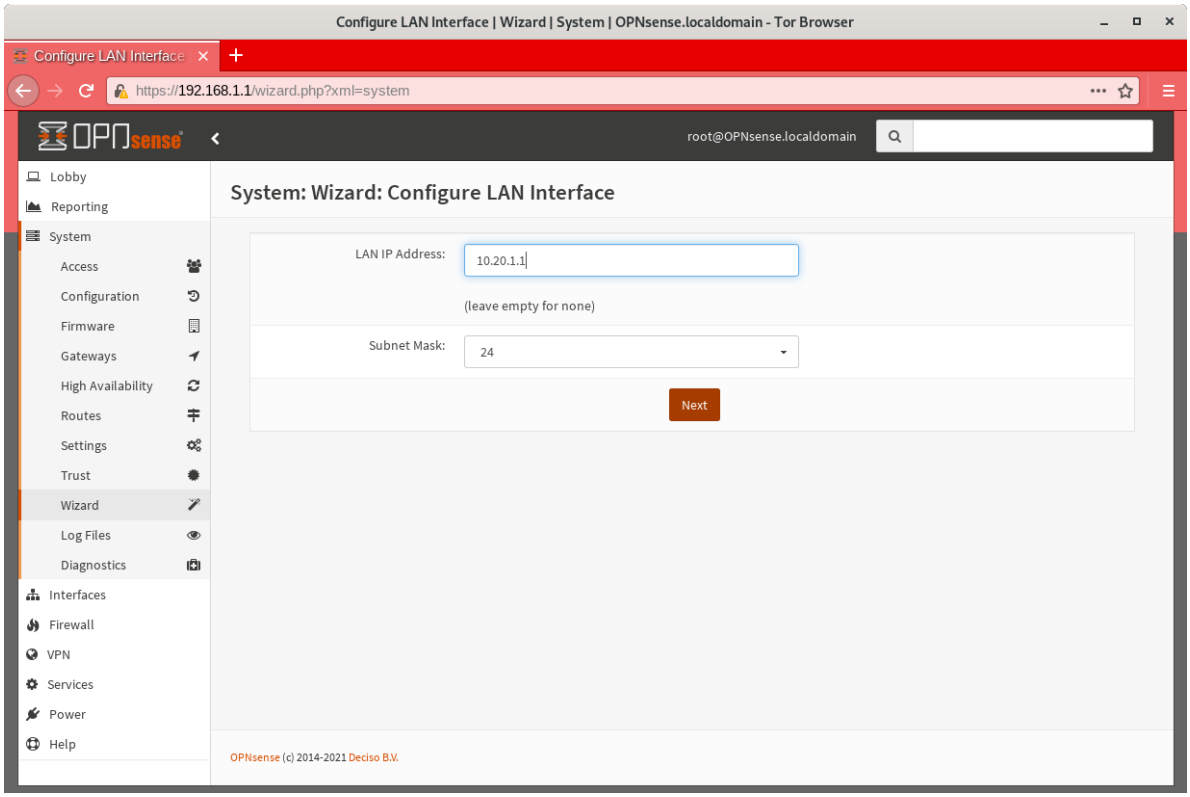


2. **Time Server Information**: Leave the default settings unchanged and click **Next**.

3. **Configure WAN Interface**: Enter the appropriate configuration for your network. Consult your local sysadmin if you are unsure what to enter here. For many environments, the default of DHCP will work and the rest of the fields can be left at their default values.

   Click **Next** to proceed.

4. **Configure LAN Interface**: Use the IP address of the *Admin Gateway* (`10.20.1.1`) and the subnet mask (`/24`) of the *Admin Subnet*. Click **Next**.



5. **Set Root Password**: If the password was already reset during the 2FA setup, you don't need to set it again. If it was not, then set a strong password now and store it in the *Admin Workstation*'s KeePassXC database. Click **Next** to continue.

6. **Reload Configuration**: Click **Reload** to apply the changes you made in the Setup Wizard.

At this point, since the LAN subnet settings were changed from their defaults, you will no longer be able to connect after reloading the firewall and the reload will time out. This is not an error - the firewall has reloaded and is working correctly.

To connect to the new LAN interface, unplug and reconnect your network cable to get a new network address assigned via DHCP. Note that if you used a subnet with fewer addresses than `/24`, the default DHCP configuration in OPNSense may not work. In this case, you should assign the Admin Workstation a static IP address that is known to be in the subnet to continue.

The Web GUI will now be available on the *Admin Gateway* IP address. Navigate to `https://<Admin Gateway IP>` in the *Unsafe Browser* and log in to the `root` account using an OTP token and the passphrase you just set.

Once you've logged in to the Web GUI, you are ready to continue configuring the firewall.

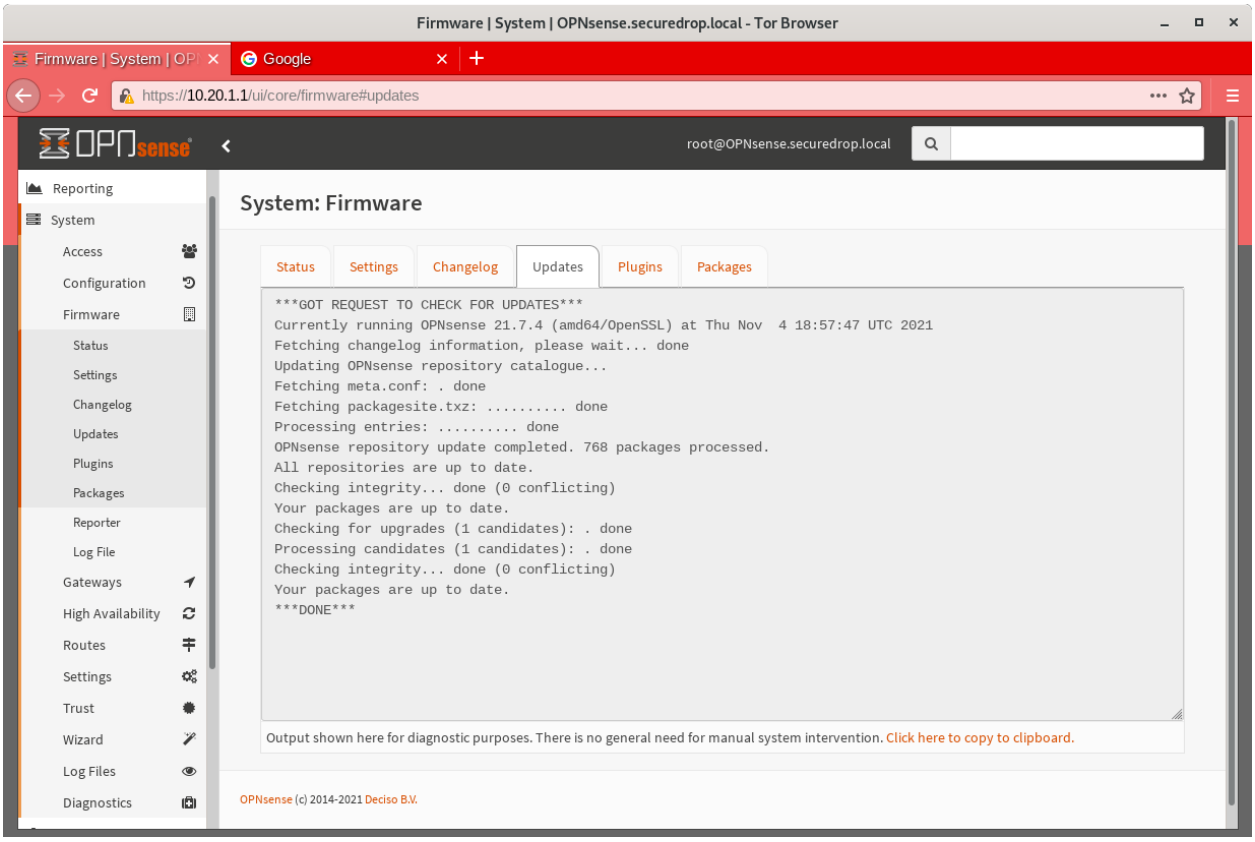
Connect interfaces and test

Now that the initial configuration is completed, you can connect the WAN port without potentially conflicting with the default LAN settings (as explained earlier). Connect the WAN port to the external network. You can watch the WAN entry in the Interfaces table on the OPNSense Dashboard homepage to see as it changes from down (red arrow pointing down) to up (green arrow pointing up). This usually takes several seconds. The WAN's IP address will be shown once it comes up.

Finally, test connectivity to make sure you are able to connect to the Internet through the WAN. The easiest way to do this is to open another tab in the Unsafe Browser and visit a host that you expect to be up (e.g. `google.com`).

Update OPNSense to the latest version

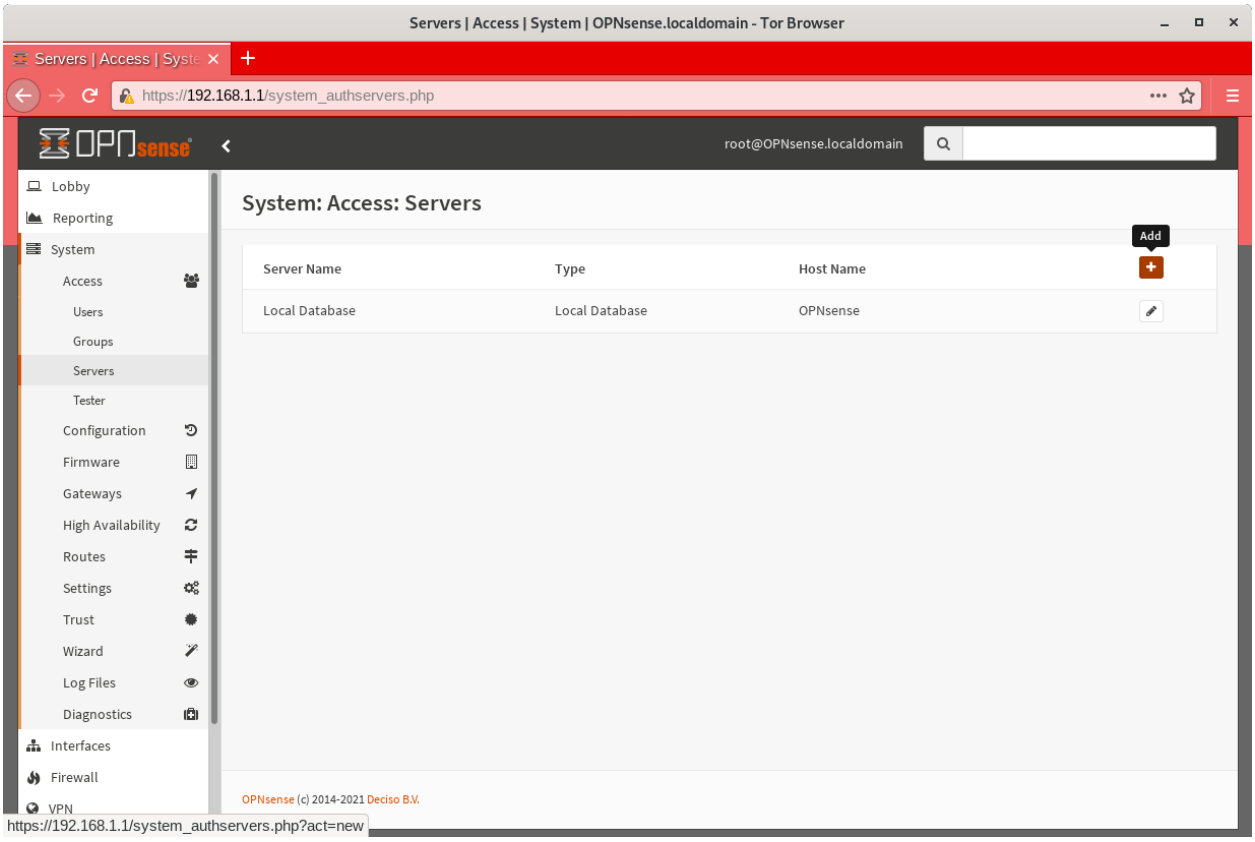
You should update OPNSense to the latest version available before proceeding with the rest of the configuration. Navigate to **Lobby ► Dashboard** and click **Click to check for updates** to start the process, and follow any on-screen instructions to complete the update. Note that a reboot may be required, and you may also need to apply several updates in a row to get to the latest version.



Enable *Two-Factor Authentication*

OPNSense supports *Two-Factor Authentication* (2FA) via mobile apps such as Google Authenticator or FreeOTP. To set it up, first make sure you have a mobile device available with your choice of 2FA app.

Next, in the OPNSense Web GUI, navigate to **System ► Access ► Servers** and click **+** to add a new server.

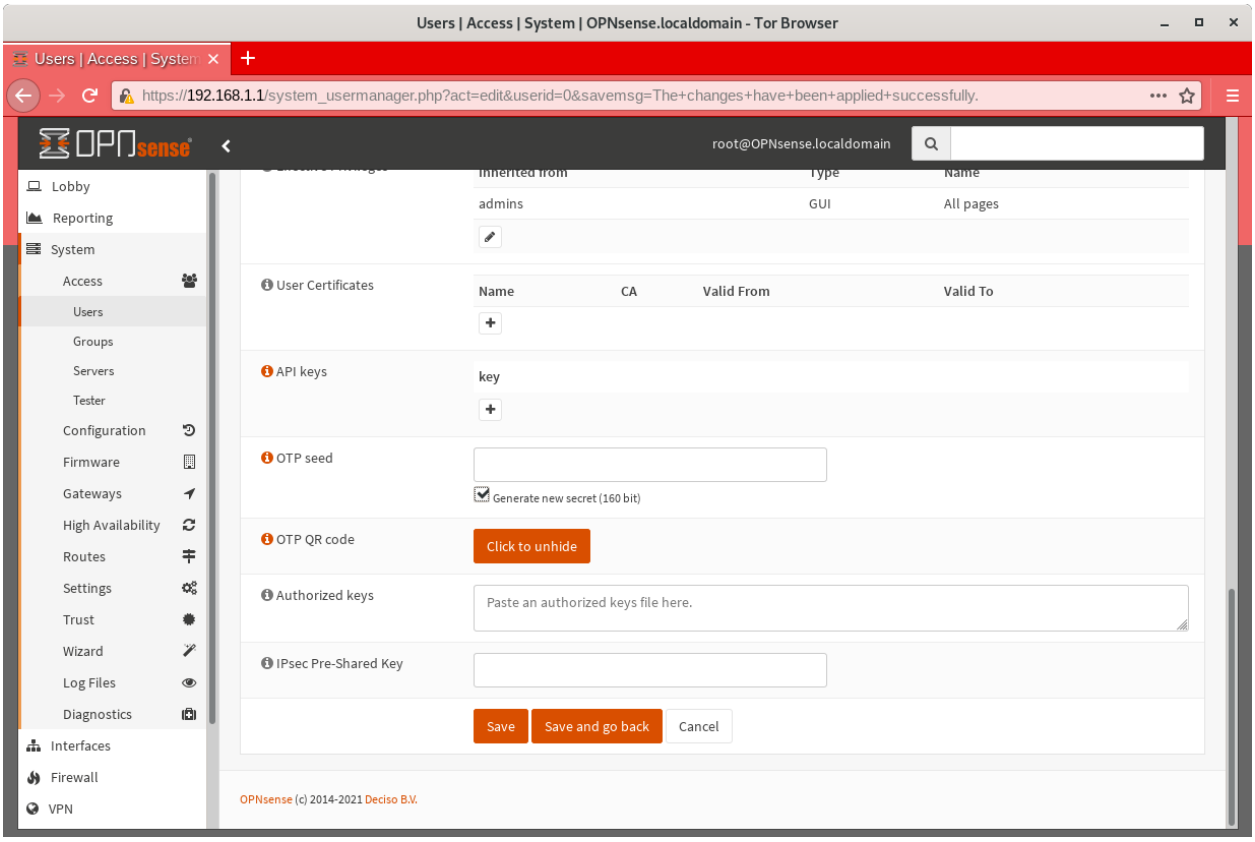


Note

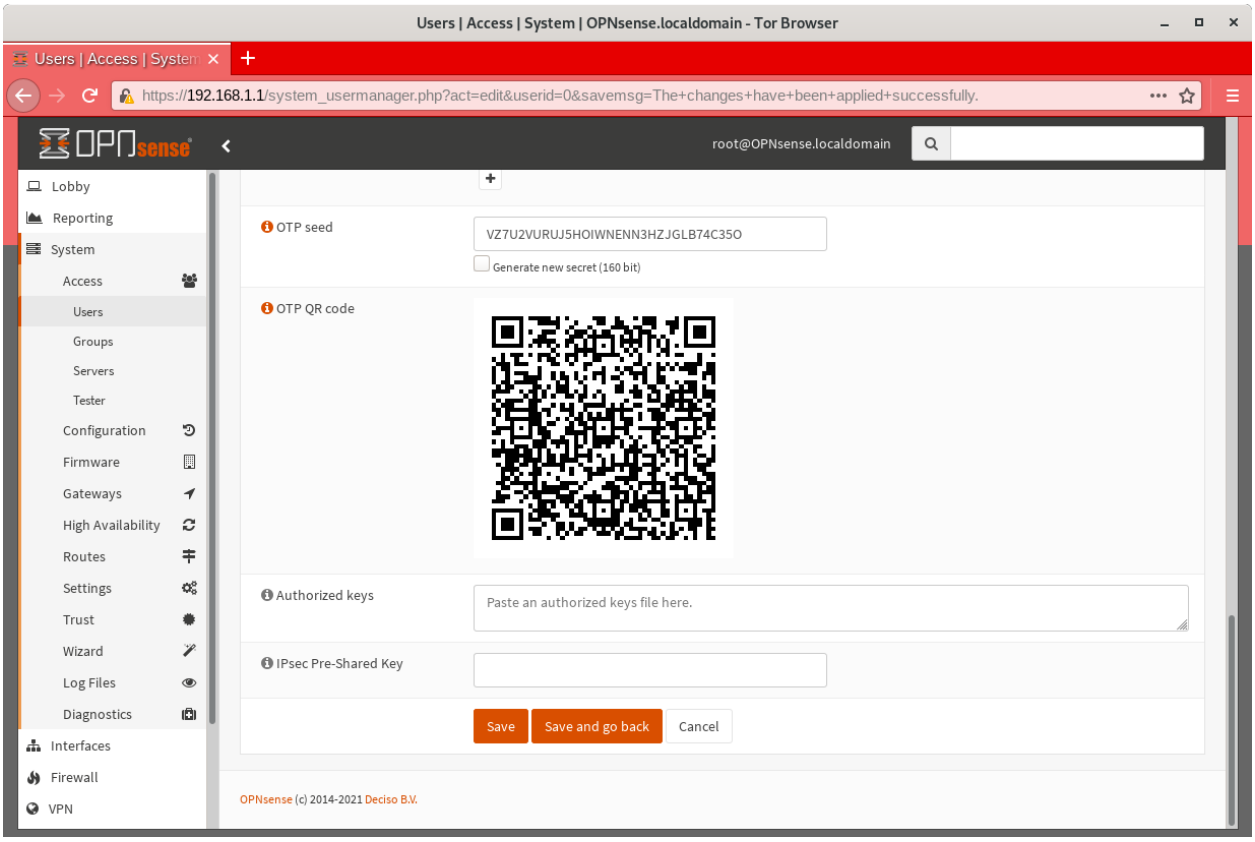
The time on your firewall must be set correctly for 2FA to work properly. This should happen automatically once the WAN connection is established.

On the next page, enter `TOTP Local` in the **Descriptive name** field and choose `Local + Timebased One Time Password` from the **Type** dropdown. Leave the other fields at their default values and click **Save**

Next, navigate to **System ► Access ► Users** and click the edit button for the `root` user. Scroll down the page to the **OTP seed** section and check the **Generate new secret (160bit)** checkbox. Finally, click **Save**.



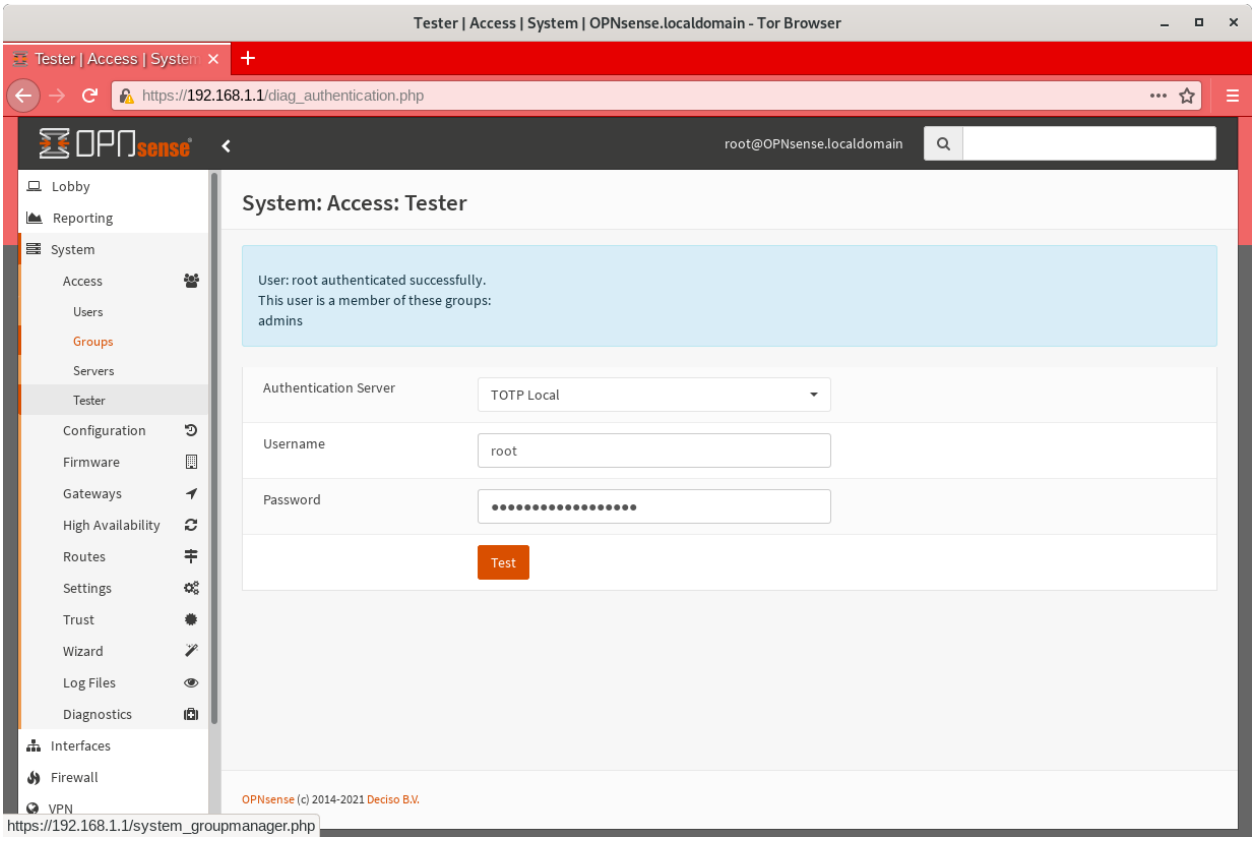
Once the page has reloaded, scroll down to the **OTP QR code** section and click **Click to unhide**, then scan the generated QR code with your mobile auth application of choice.



If you wish, you may also save the OTP seed value displayed above the QR code in your Tails KeePassXC database - this isn't required, but will allow you to set up TOTP on another mobile device if you need to in the future.

Test your new login credentials

To verify that your new password and OTP secret are working, navigate to **System ► Access ► Tester**. Select `TOTP Local` from the **Authentication Server** dropdown, enter the `root` username in the **Username** field, and enter your OTP token and password concatenated like `123456PASSWORD` in the **Password** field. Then click **Test**.



If the test fails, make sure you have used the correct OTP code and password, and edit the `root` user record as necessary.

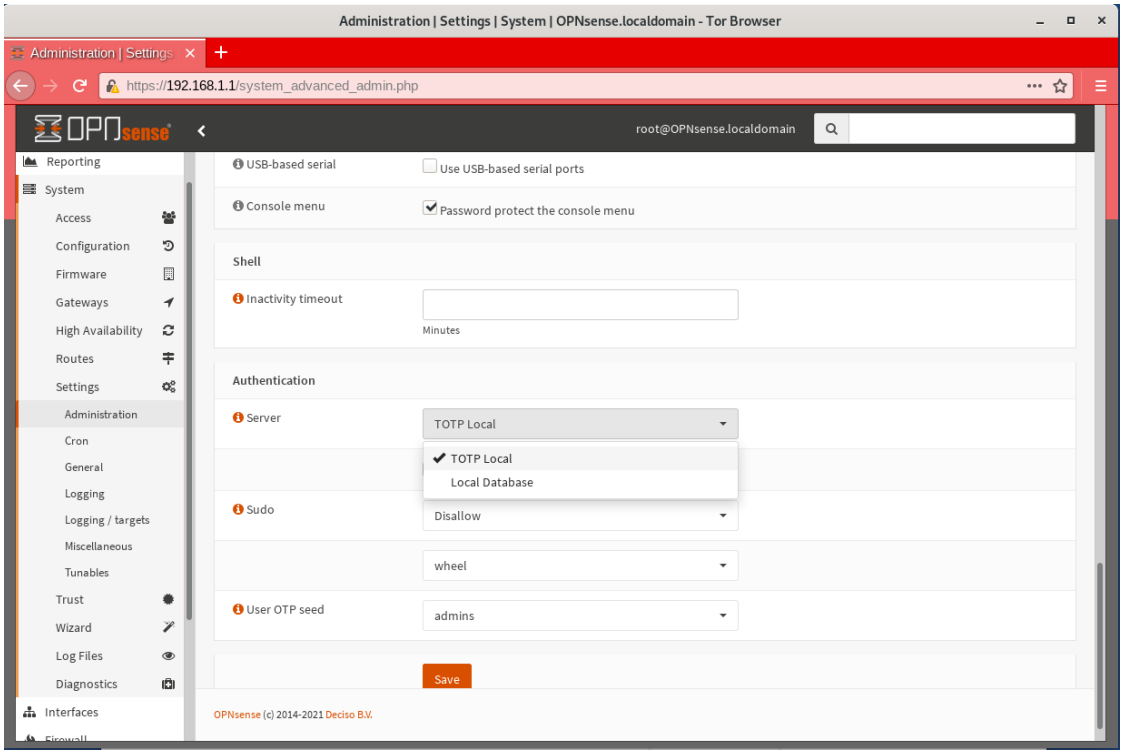
Note

You must enter the OTP token and passphrase concatenated as a single string like `123456PASSWORD` in the **Password** field.

Warning

Do not skip this test, or proceed further until it passes, as you will be locked out of the firewall Web GUI and console if the account is not set up correctly!

Finally, navigate to **System ▶ Settings ▶ Administration** and scroll down to the **Authentication** section at the bottom of the page. In the **Server** dropdown, select `TOTP Local` and deselect `Local Database.`. Click **Save**.



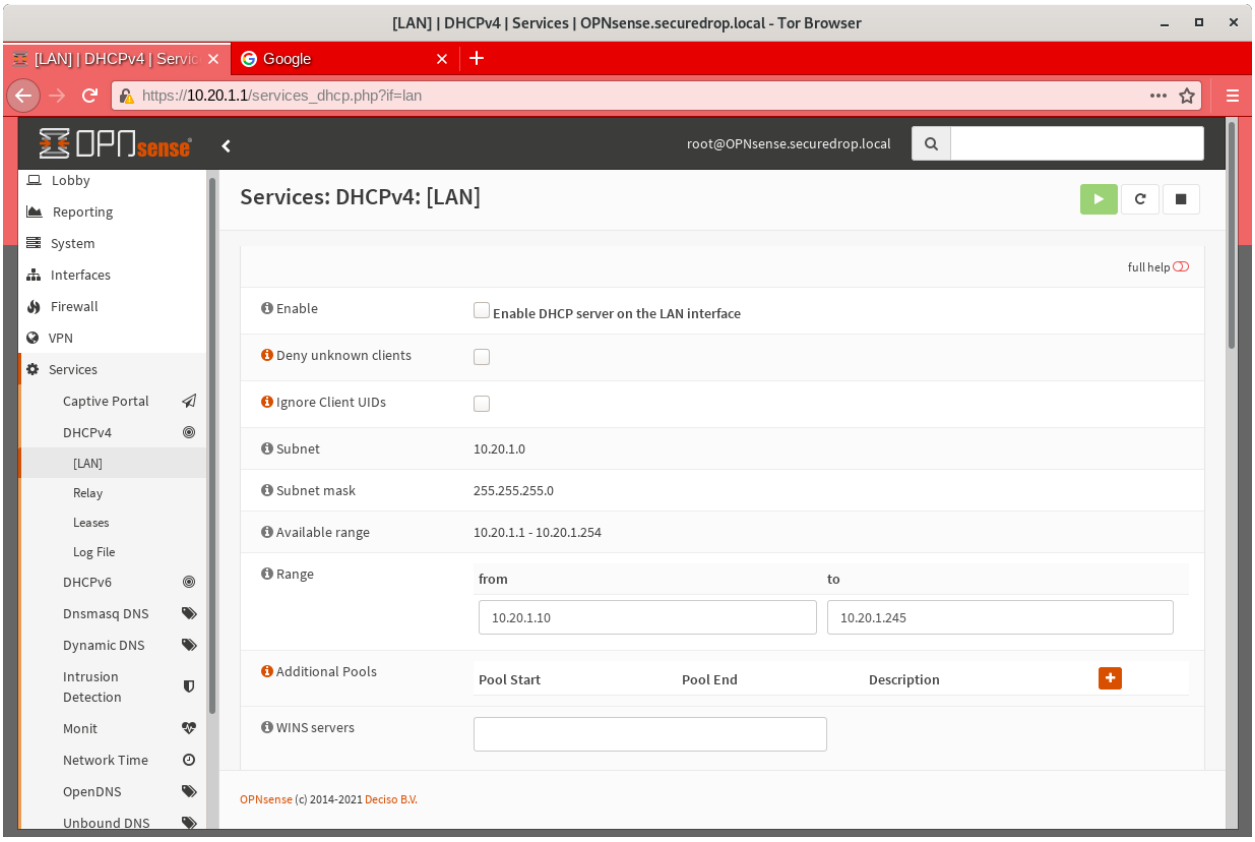
1.25.3 Disable DHCP on the firewall

OPNSense runs a DHCP server on the LAN interface by default. At this stage in the documentation, the *Admin Workstation* likely has an IP address assigned via that DHCP server.

In order to tighten the firewall rules as much as possible, we recommend disabling the DHCP server and assigning a static IP address to the Admin Workstation instead.

Disable DHCP server on the LAN interface

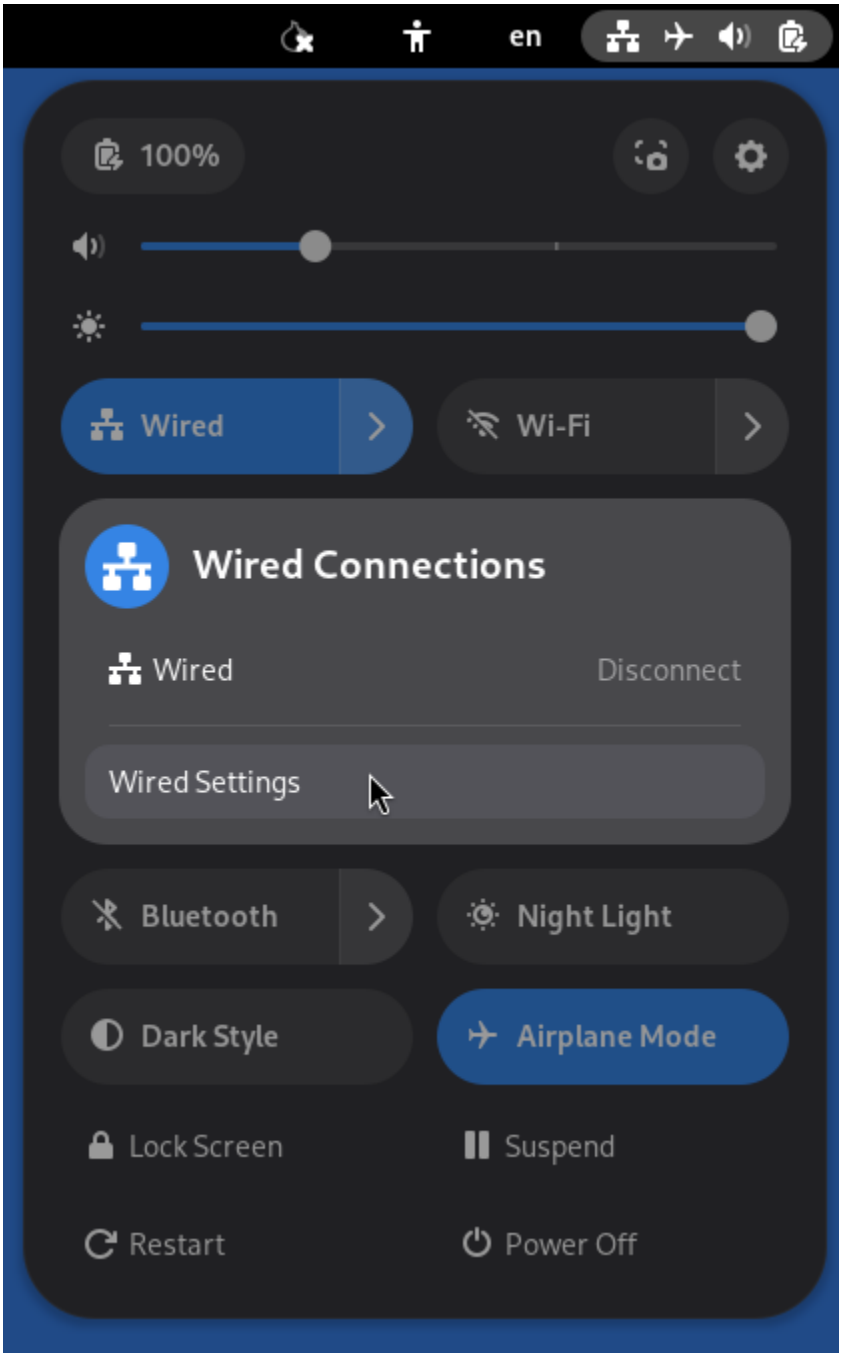
To disable DHCP, navigate to **Services ▶ DHCPv4 ▶ [LAN]** in the Web GUI. Uncheck the **Enable DHCP server on the LAN interface** checkbox, scroll down, and click **Save**.



Assign a static IP address to the *Admin Workstation*

Now you will need to assign a static IP to the *Admin Workstation*.

You can easily check your current IP address by *clicking* the top right of the menu bar, clicking on the **Wired Connection** and then clicking **Wired Settings**.



From here you can click on the cog beside the wired network connection:

This will take you to the network settings. Change to the **IPv4** tab. Ensure that **IPv4 Method** is set to **Manual**, and that the **Automatic** switch for **DNS** is in the "off" position, as highlighted in the screenshot below:

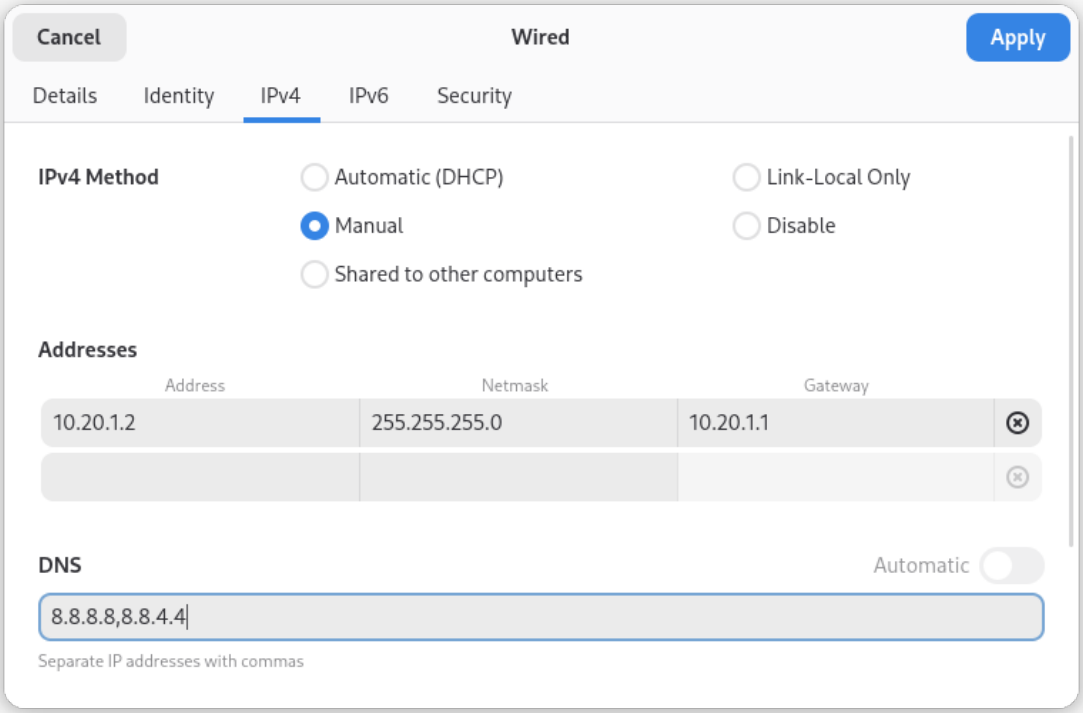
Note

The Unsafe Browser will not launch when using a manual network configuration if it does not have DNS servers configured. This is technically unnecessary for our use case because we are only using it to access IP addresses on the LAN, and do not need to resolve anything with DNS. Nonetheless, you should configure some DNS servers here so you can continue to use the Unsafe Browser to access the WebGUI in future sessions.

We recommend keeping it simple and using the same DNS servers that you used for the network firewall in the setup wizard.

Fill in the static networking information for the *Admin Workstation*:

- Address: `10.20.1.2`
- Netmask: `255.255.255.0`
- Gateway : `10.20.1.1`



Click **Apply**. If the network does not come up within 15 seconds or so, try disconnecting and reconnecting your network cable to trigger the change. You will need you have succeeded in connecting with your new static IP when you are able to connect using the Tor Connection assistant, and you see the message "Connected to Tor successfully".

Troubleshooting: DNS servers and the Unsafe Browser

After saving the new network configuration, you may still encounter the "No DNS servers configured" error when trying to launch the Unsafe Browser. If you encounter this issue, you can resolve it by disconnecting from the network and then reconnecting, which causes the network configuration to be reloaded.

To do this, click the network icon in the system toolbar, and click **Disconnect** under the name of the currently active network connection, which is displayed in bold. After it disconnects, click the network icon again and click the name of the connection to reconnect. You should see a popup notification that says "Connection Established", and the Tor Connection assistant should show the message "Connected to Tor successfully".

For the next step, SecureDrop Configuration, you will manually configure the firewall for SecureDrop, using screenshots as a reference.

1.25.4 SecureDrop configuration

SecureDrop uses the firewall to achieve two primary goals:

1. Isolating SecureDrop from the existing network, which may be compromised (especially if it is a venerable network in a large organization like a newsroom).
2. Isolating the *Application Server* and the *Monitor Server* from each other as much as possible, to reduce attack surface.

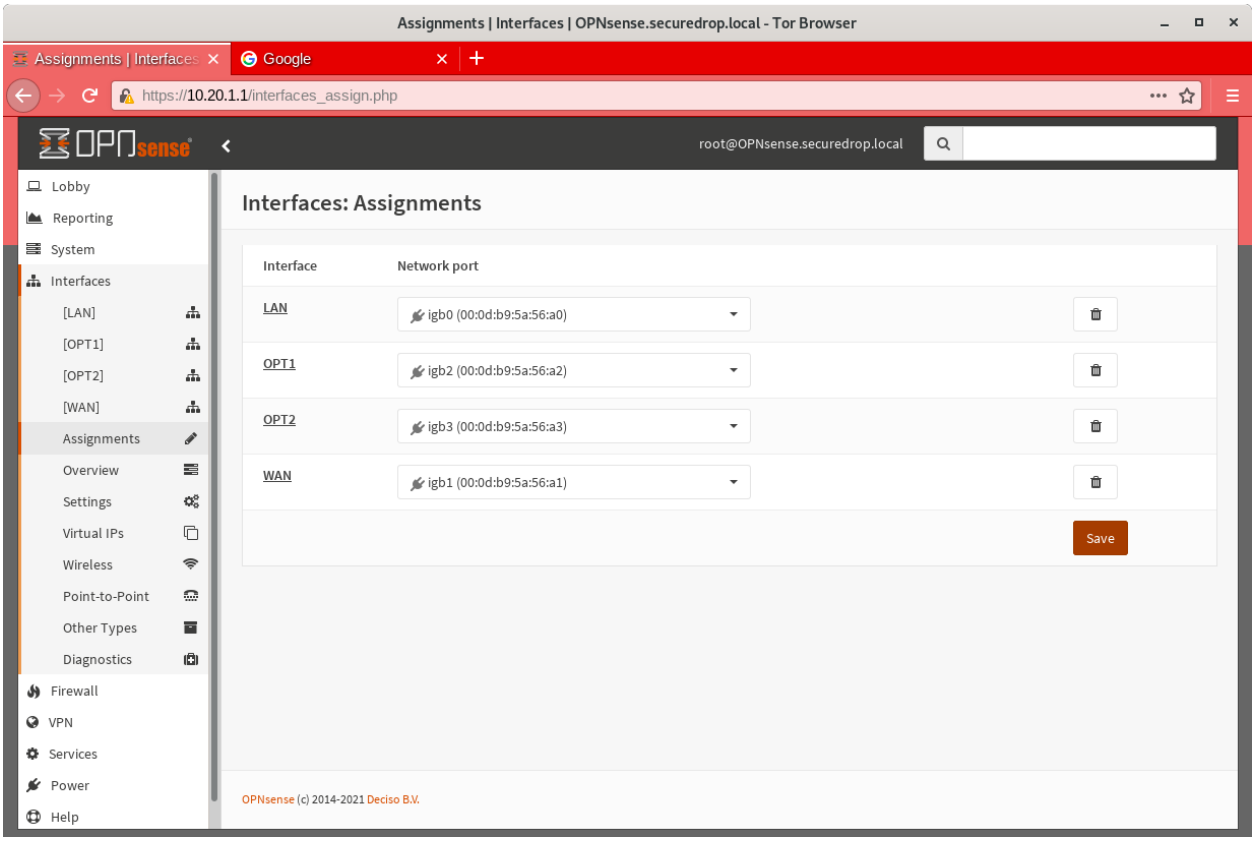
In order to use the firewall to isolate the *Application Server* and the *Monitor Server* from each other, we need to connect them to separate interfaces, and then set up firewall rules that allow them to communicate.

Enable the OPT1 and OPT2 interfaces

The OPT1 and OPT2 interfaces will be used for the *Application Server* and *Monitor Server* respectively. To enable them, first connect the *Application Server* to the physical OPT1 port and the *Monitor Server* to the OPT2 port.

Next, navigate to **Interfaces ▶ Assignments**. LAN and WAN will already be enabled. Click the **+** button in the **New Interface** section to enable the OPT1 interface on the next available NIC (`igb2` in the screenshot below). Once OPT1 has been added, click **+** again to add OPT2 (on `igb3` in the screenshot below)

Finally, click **Save**.



Configure the LAN, WAN, OPT1, and OPT2 interfaces

OPT1 and OPT2 need to be configured to use the subnets defined for the *Application* and *Monitor Servers*, and some additional configuration is required for the LAN and WAN interfaces, that is not covered by the Setup Wizard.

Configure the WAN interface

First, navigate to **Interfaces ► [WAN]**. In the **Basic configuration** section, check the checkbox labeled **Prevent interface removal**.

In the **Generic configuration** section, make sure that the **Block private networks** and **Block bogon networks** checkboxes are checked.

Scroll down and click **Save**, then click **Apply changes** when prompted.

Configure the LAN interface

Next, navigate to **Interfaces ► [LAN]**. In the **Basic configuration** section, check the checkbox labeled **Prevent interface removal**.

In the **Generic configuration** section, select `Static IPv4` in the **IPv4 Configuration Type** dropdown, and `None` in the **IPV6 Configuration Type** dropdown.

Scroll down and click **Save**, then click **Apply changes** when prompted.

Configure the OPT1 interface

Next, navigate to **Interfaces ► [OPT1]**. In the **Basic configuration** section, check the checkboxes labeled **Enable interface** and **Prevent interface removal**.

In the **Generic configuration** section, select `Static IPv4` in the **IPv4 Configuration Type** dropdown, and `None` in the **IPV6 Configuration Type** dropdown.

Scroll down. In the **Static IPv4 Configuration** section, enter the *Application Gateway* IP address and routing prefix (`10.20.2.1` and `24` if you are using the recommended values).

Click **Save**, then click **Apply changes** when prompted.

Configure the OPT2 interface

Finally, navigate to **Interfaces ► [OPT2]**. In the **Basic configuration** section, check the checkboxes labeled **Enable interface** and **Prevent interface removal**.

In the **Generic configuration** section, select `Static IPv4` in the **IPv4 Configuration Type** dropdown, and `None` in the **IPV6 Configuration Type** dropdown.

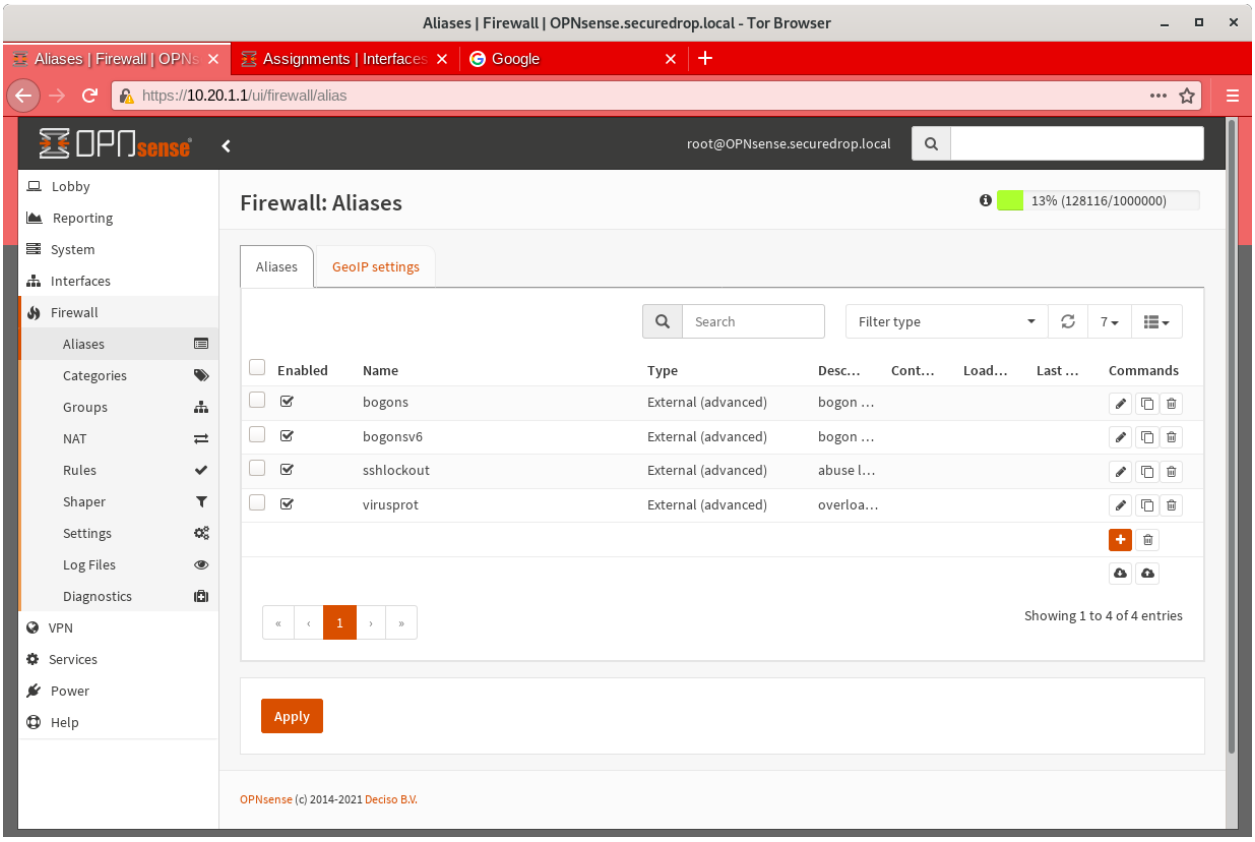
Scroll down. In the **Static IPv4 Configuration** section, enter the *Monitor Gateway* IP address and routing prefix (`10.20.3.1` and `24` if you are using the recommended values).

Click **Save**, then click **Apply changes** when prompted.

Configure firewall aliases

In order to simplify firewall rule setup, the next step is to configure aliases for hosts and ports referred to in the rules.

To start, first navigate to **Firewall ► Aliases**. You should see some system-defined aliases as shown below:

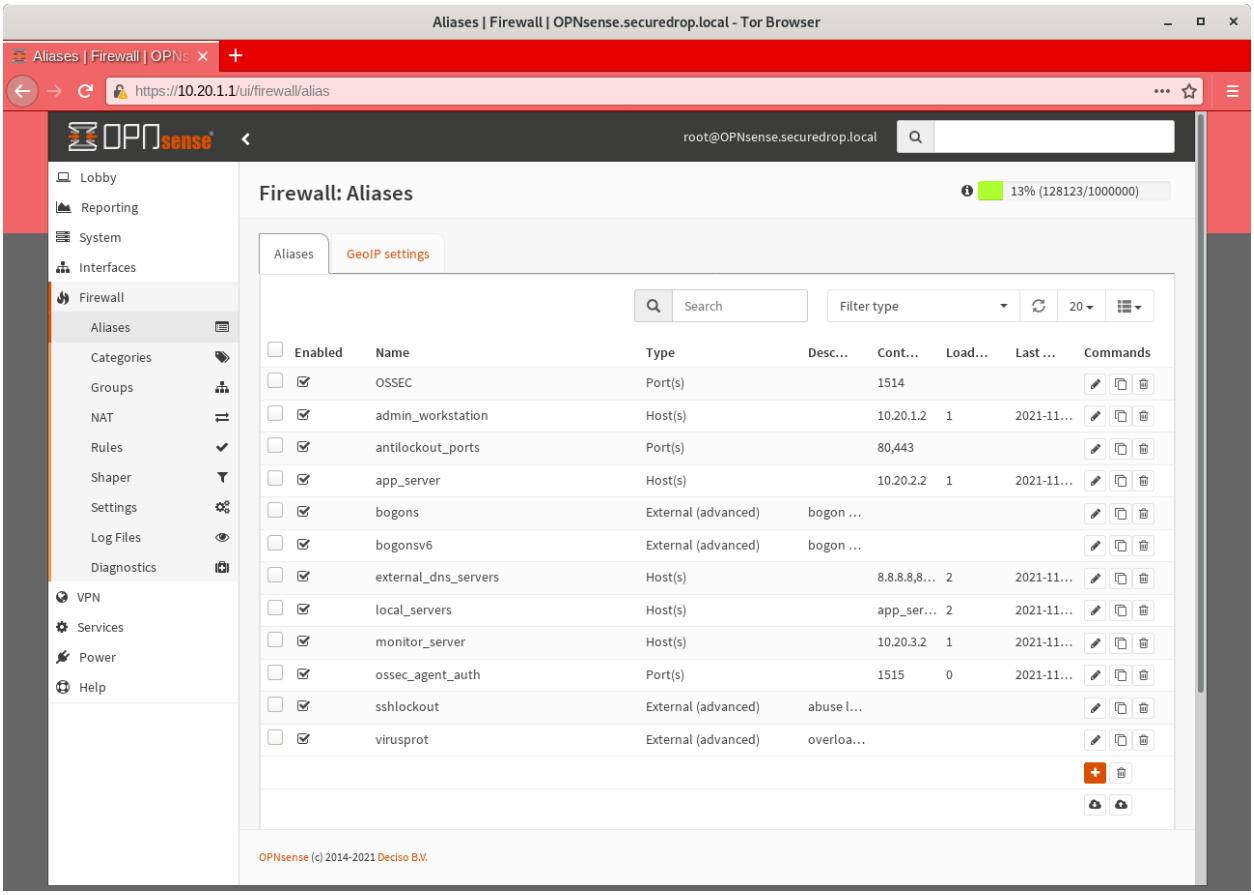


Click the **+** button to add new aliases. You should add the aliases defined in the table below (assuming recommended values for IP addresses):

Table 1: Firewall Aliases

Name	Type	Content
admin_workstation	Host(s)	`10.20.1.2`
app_server	Host(s)	`10.20.2.2`
external_dns_servers	Host(s)	`8.8.8.8, 8.8.4.4`
monitor_server	Host(s)	`10.20.3.2`
local_servers	Host(s)	`app_server, monitor_server`
OSSEC	Port(s)	`1514`
ossec_agent_auth	Port(s)	`1515`
antilockout_ports	Port(s)	`80, 443`

When complete, the **Aliases** page should look like this:



Scroll down and click **Apply** to save and apply your new aliases.

Configure firewall rules

Next, configure firewall rules for each interface.

Configure firewall rules on LAN

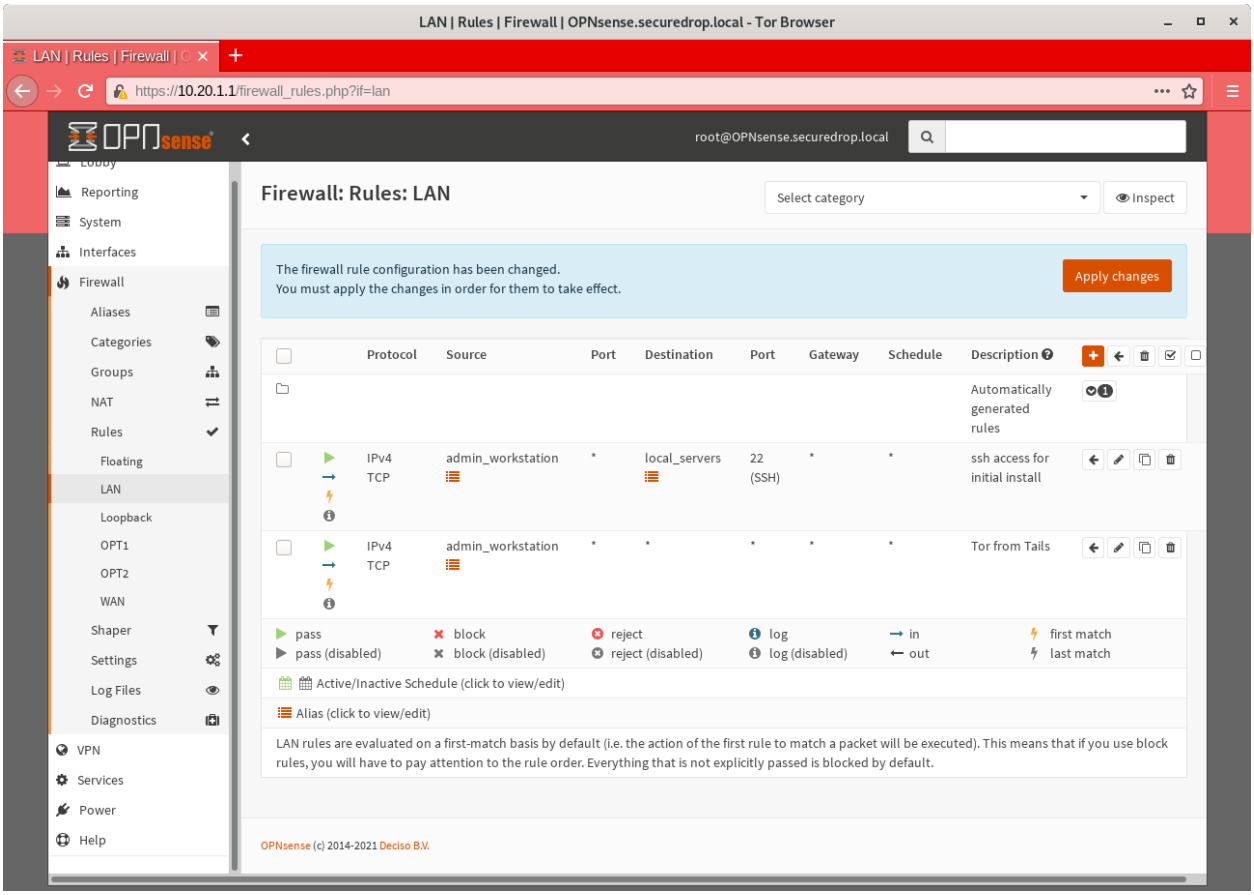
First, navigate to **Firewall ► Rules ► LAN**. The LAN interface should have one automatically-generated anti-lockout rule in place, in addition to two default-allow rules. The default-allow rules should be removed once the SecureDrop-specific rules below have been added. The anti-lockout feature should be disabled as a last step.

The rules needed are described in this table:

Table 2: Firewall Rules - LAN

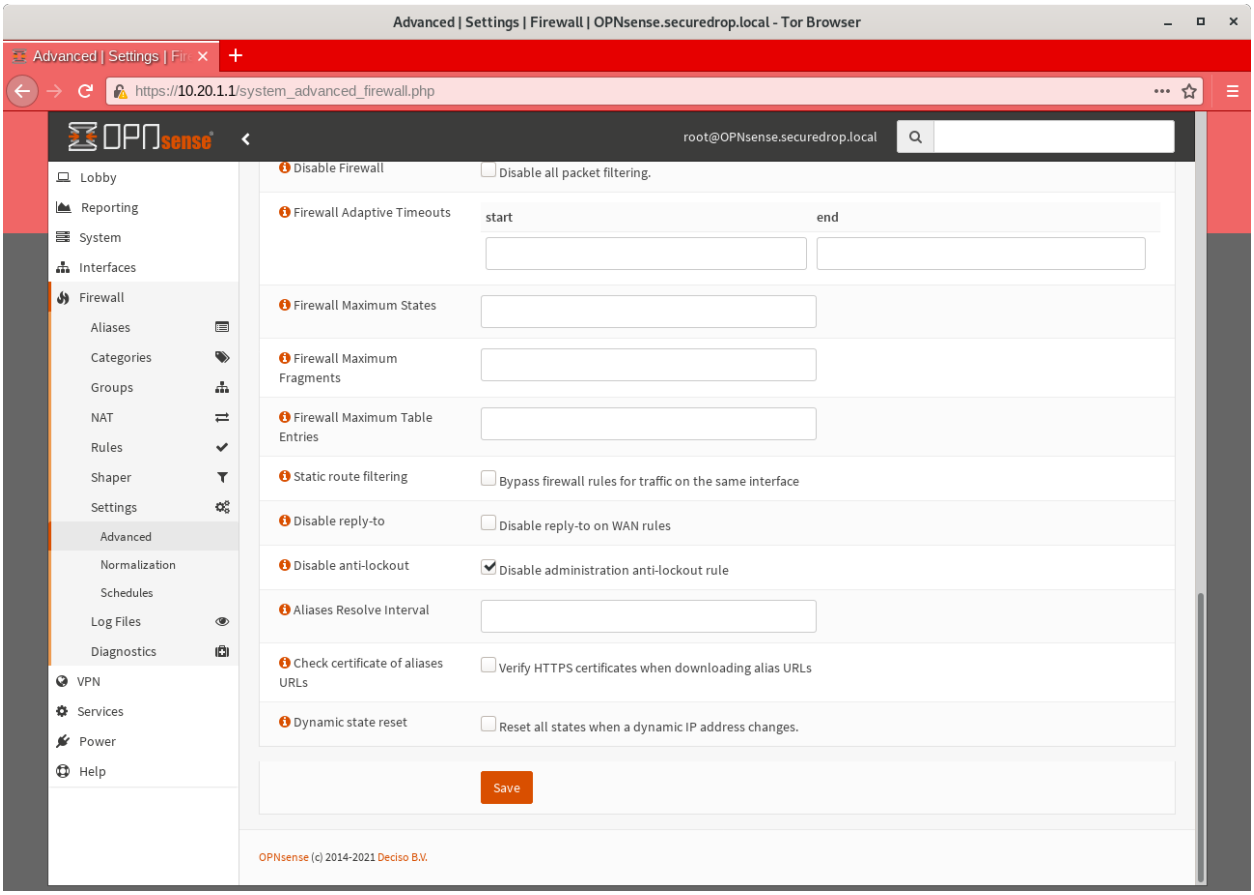
Action	TCP/IP Version	Protocol	Src	Src port	Dest	Dest port	Description
Pass	IPv4	TCP	admin_workstat	•	local_servers	22 (SSH)	SSH access for initial install
Pass	IPv4	TCP	admin_workstat	•	•	•	Tor from Tails

Add or remove rules until they match the following screenshot including ordering. Click the **+** button to add a rule.



Once the rules match, click **Apply Changes.**

Finally, remove the default anti-lockout rule. First, navigate to **Firewall ► Settings ► Advanced**. Scroll down to the **Miscellaneous** section and check the **Disable anti-lockout** checkbox. Then, click **Save**.



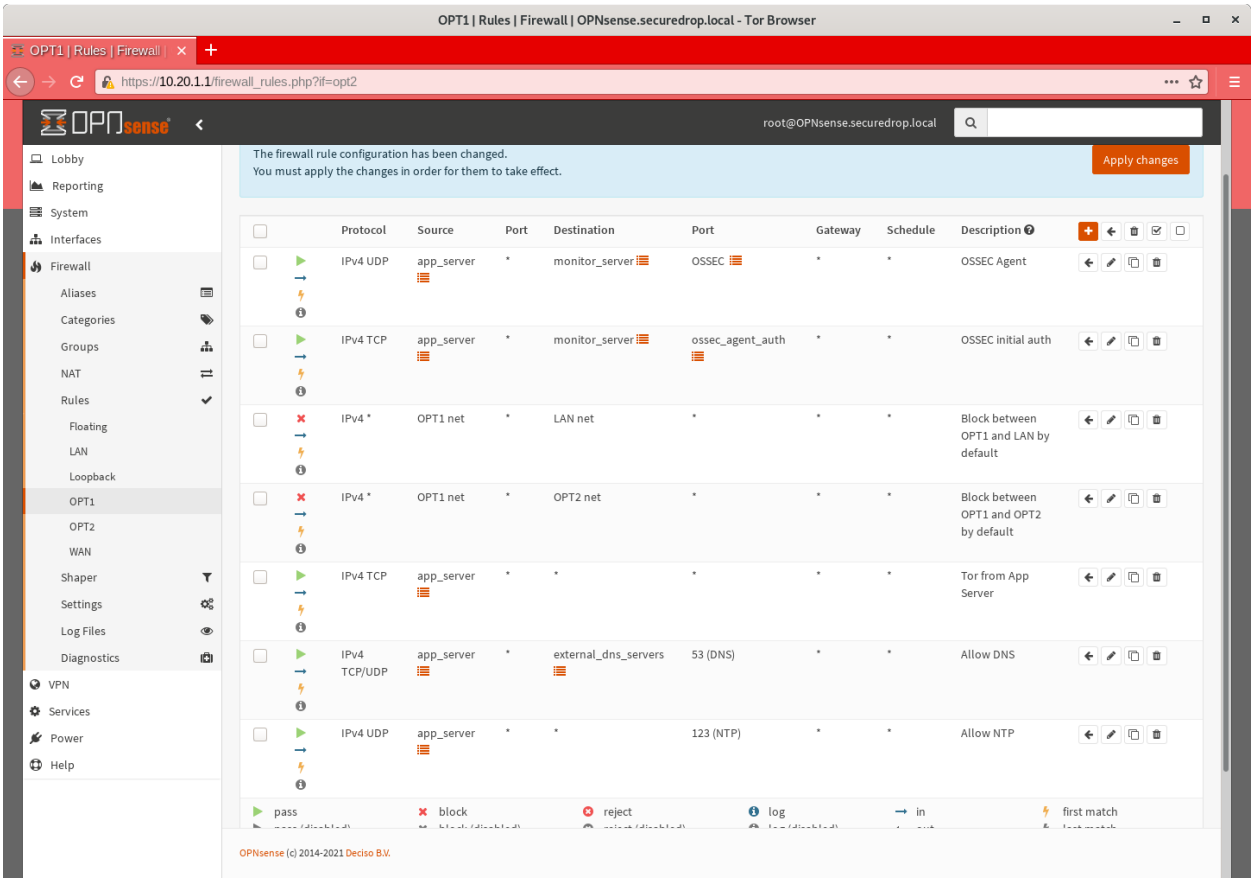
Configure firewall rules on OPT1

Next, navigate to **Firewall ▶ Rules ▶ OPT1**. There should be no rules defined on this interface. Add the rules below:

Table 3: Firewall Rules - OPT1

Action	TCP/IP Version	Protocol	Src	Src port	Dest	Dest port	Description
Pass	IPv4	UDP	app_server	*	monitor_server	OSSEC	OSSEC Agent
Pass	IPv4	TCP	app_server	*	monitor_server	ossec_agent_au	OSSEC initial auth
Block	IPv4	any	OPT1 net	*	LAN net	*	Block between OPT1 and LAN by default
Block	IPv4	any	OPT1 net	*	OPT2 net	*	Block between OPT1 and OPT2 by default
Pass	IPv4	TCP	app_server	*	*	*	Tor from App Server
Pass	IPv4	TCP/UDP	app_server	*	external_dns_serv	53 (DNS)	Allow DNS
Pass	IPv4	UDP	app_server	*	*	123 (NTP)	Allow NTP

Once they match the screenshot below, click **Apply Changes**.

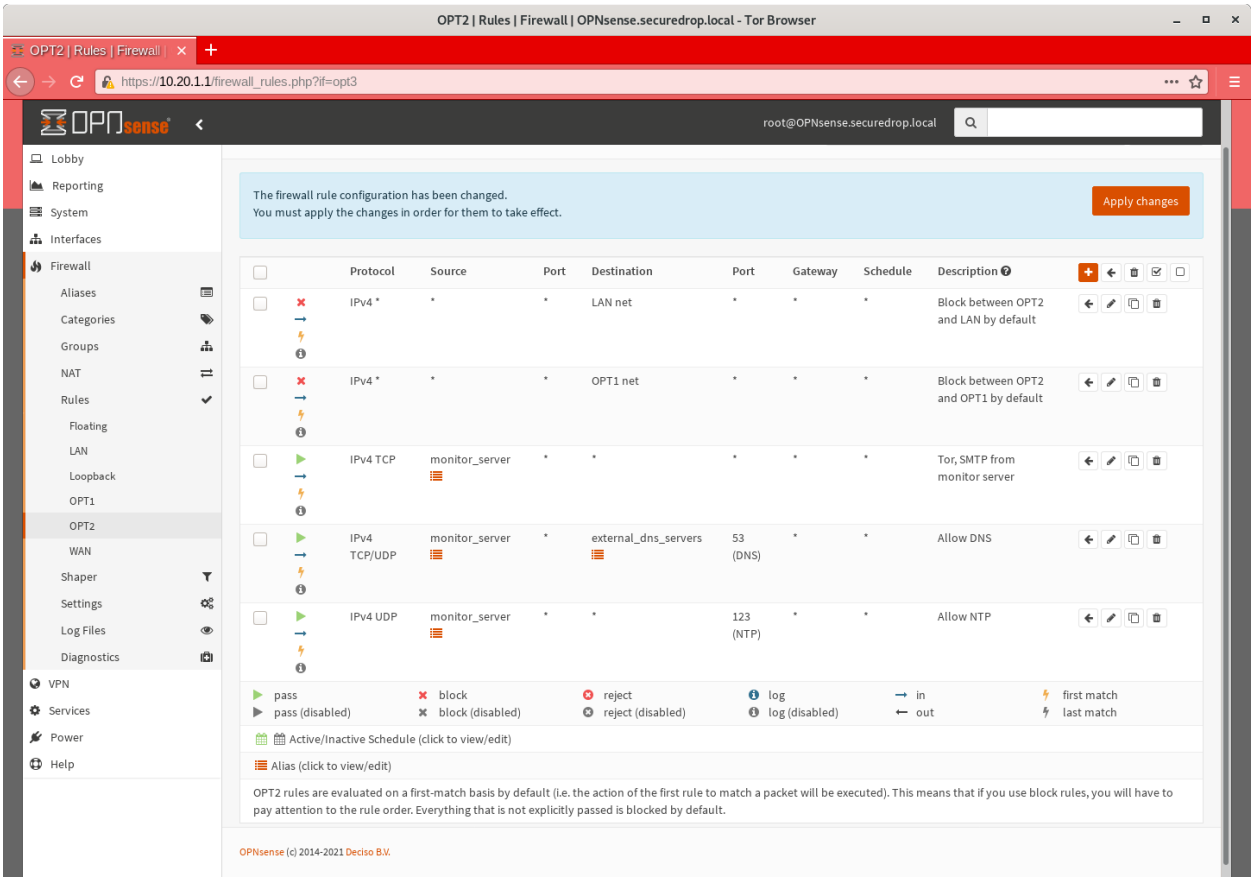


Configure firewall rules on OPT2

Next, navigate to **Firewall ► Rules ► OPT2**. Similarly to OPT1, there should be no rules defined on this interface. Add the rules below until the rules in the Web GUI match those in the screenshot:

Table 4: Firewall Rules - OPT2

Action	TCP/IP Version	Protocol	Src	Src port	Dest	Dest port	Description
Block	IPv4	any	OPT2 net	*	LAN net	*	Block between OPT2 and LAN by default
Block	IPv4	any	OPT2 net	*	OPT1 net	*	Block between OPT2 and OPT1 by default
Pass	IPv4	TCP	monitor_server	*	*	*	Tor, SMTP from Monitor Server
Pass	IPv4	TCP/UDP	monitor_server	*	external_dns_serv	53 (DNS)	Allow DNS
Pass	IPv4	UDP	monitor_server	*	*	123 (NTP)	Allow NTP



Finally, click **Apply Changes**.

The *Network Firewall* configuration is now complete, allowing you to move to the next step: *setting up the servers.*

1.25.5 Troubleshooting tips

Here are some general tips for setting up OPNSense firewall rules:

1. Create aliases for the repeated values (IPs and ports).
2. OPNSense is a stateful firewall, which means that you don't need corresponding rules to allow incoming traffic in response to outgoing traffic (like you would in, e.g. iptables with `--state ESTABLISHED,RELATED`).
3. You should create the rules *on the interface where the traffic originates*.
4. Make sure you delete the default "allow all" rule on the LAN interface.
5. If you are troubleshooting connectivity, the firewall logs can be very helpful. You can find them in the Web GUI in **Firewall ▶ Log Files**

1.25.6 Keeping OPNSense up to date

Periodically, the OPNSense project maintainers release an update to the OPNSense software running on your firewall. You can check for updates using the link on the OPNSense dashboard.

If you see that an update is available, we recommend installing it. Most of these updates are for minor bugfixes, but occasionally they can contain important security fixes. You should keep apprised of updates yourself by checking the OPNSense Blog or subscribing to the OPNSense Blog RSS feed.

1.26 Prepare the servers

1.26.1 Pre-install steps

Upgrade the server BIOS

Before beginning the installation process, you should upgrade your servers' BIOS to the most recent stable version available. This process will differ for each server make/model - if you are using one of the recommended NUC models, you can find instructions in *BIOS updates on the servers*.

Update BIOS settings

Once the BIOS has been updated, you should boot into it again to disable any unused hardware, including:

- wireless LAN and Bluetooth
- Thunderbolt support
- audio support (output, speakers, microphones)
- other features supported by the hardware but not used by SecureDrop, such as Thunderbolt, SD card controller, or enhanced consumer infrared

In most cases, you should enable support for LAN and USB ports only. On NUC models, you can find this under **Advanced ▶ Onboard Devices**

You should also check the servers' boot settings. Ubuntu 24.04 supports both Legacy and UEFI boot modes, with UEFI preferred. You should also disable Secure Boot. SecureDrop uses a custom kernel with security patches, which is unsigned and will not boot if Secure Boot is enabled.

Our *specific hardware recommendations* enumerate recommended BIOS settings for hardware that we have tested.

1.26.2 Install Ubuntu

The SecureDrop *Application Server* and *Monitor Server* run **Ubuntu 24.04.3 LTS (Noble Numbat)**. To install Ubuntu on the servers, you must first download and verify the Ubuntu installation media.

Ubuntu introduction

Note

Installing Ubuntu is simple and may even be something you are very familiar with, but it is **strongly** encouraged that you read and follow this documentation exactly as there are some "gotchas" that may cause your SecureDrop setup to break.

The SecureDrop *Application Server* and *Monitor Server* run **Ubuntu Server 24.04.3 LTS (Noble Numbat)**. To install Ubuntu on the servers, you must first download and verify the Ubuntu installation media.

Download the Ubuntu installation media

The installation media and the files required to verify it are available on the Ubuntu Releases page. You will need to download the following files:

- ubuntu-24.04.3-live-server-amd64.iso
- SHA256SUMS
- SHA256SUMS.gpg

Alternatively, you can use the command line:

```
cd ~/Downloads
curl -OOO https://releases.ubuntu.com/24.04.3/{ubuntu-24.04.3-live-server-amd64.iso,
↪SHA256SUMS{,.gpg}}
```

Verify the Ubuntu installation media

You should verify the Ubuntu image you downloaded hasn't been modified by a malicious attacker or otherwise corrupted. To do so, check its integrity with cryptographic signatures and hashes.

First, download both *Ubuntu Image Signing Keys* and verify their fingerprints.

```
gpg --recv-key --keyserver hkps://keyserver.ubuntu.com \
"C598 6B4F 1257 FFA8 6632 CBA7 4618 1433 FBB7 5451" \
"8439 38DF 228D 22F7 B374 2BC0 D94A A3F0 EFE2 1092"
```

Note

It is important you type this out correctly. If you are not copy-pasting this command, double-check you have entered it correctly before pressing enter.

Again, when passing the full public key fingerprint to the `--recv-key` command, GPG will implicitly verify that the fingerprint of the key received matches the argument passed.

Caution

If GPG warns you that the fingerprint of the key received does not match the one requested **do not** proceed with the installation. If this happens, please email us at securedrop@freedom.press.

Next, verify the `SHA256SUMS` file.

```
gpg --keyid-format long --verify SHA256SUMS.gpg SHA256SUMS
```

Move on to the next step if you see "Good Signature" in the output, as below. Note that any other message (such as "Can't check signature: no public key") means that you are not ready to proceed.

```
gpg: Signature made Thu 11 Feb 2021 02:07:58 PM EST
gpg:                using RSA key 843938DF228D22F7B3742BC0D94AA3F0EFE21092
gpg: Good signature from "Ubuntu CD Image Automatic Signing Key (2012) <cdimage@ubuntu.
↪com>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 8439 38DF 228D 22F7 B374  2BC0 D94A A3F0 EFE2 1092
```

The next and final step is to verify the Ubuntu image.

```
sha256sum -c <(grep ubuntu-24.04.3-live-server-amd64.iso SHA256SUMS)
```

If the final verification step is successful, you should see the following output in your terminal.

```
ubuntu-24.04.3-live-server-amd64.iso: OK
```

Caution

If you do not see the line above it is not safe to proceed with the installation. If this happens, please contact us at securedrop@freedom.press.

Create the Ubuntu installation media

The Ubuntu website has detailed instructions on how to to create a bootable Ubuntu Server USB flash drive.

Follow the instructions at the link below for your operating system, then return to this page:

- Create a bootable Ubuntu USB drive on Mac
- Create a bootable Ubuntu USB drive on Windows
- Create a bootable Ubuntu USB drive on Linux

With the Ubuntu Server install USB flash drive ready, you may now proceed to the installation.

Perform the installation

The steps below are the same for both the *Application Server* and the *Monitor Server*.

Start by inserting the Ubuntu installation media into the server. Boot or reboot the server with the installation media inserted, and enter the boot menu. To enter the boot menu, you need to press a key as soon as you turn the server on. This key varies depending on server model, but common choices are Esc, F2, F10, and F12. Often, the server will

briefly display a message on boot that shows which key should be pressed to enter the boot menu. Once you've entered the boot menu, select the installation media (USB or CD) and press Enter to boot it.

On newer hardware, such as the NUC12s, you may need to use a newer Linux kernel than the one that ships by default in **Ubuntu Server 24.04.3** in order to have more up-to-date hardware drivers. To use a newer Linux kernel, select **Ubuntu Server with the HWE kernel** in the initial OS boot menu that appears prior to booting the Ubuntu image.

After booting the Ubuntu image, select **Install Ubuntu Server**.

Follow the steps to select your language, country and keyboard settings. Once that's done, let the installation process continue.

Configure the network

On the **Network connections** screen, the installer will ask you to configure at least one interface for use by the server. Your server should only have one available, corresponding to its Ethernet, usually named `eno1`. Select its list entry using the arrow keys and press **Enter**, then select **Edit IPv4** and press **Enter** again.

The **Edit eno1 IPv4 configuration** dialog will be displayed. In the **IPv4 Method** menu, select **Manual**, then add your server-specific settings.

Note

For a production install with a pfSense network firewall in place, the *Application Server* and the *Monitor Server* are on separate networks. You may choose your own network settings at this point, but make sure the settings you choose are unique on the firewall's network and remember to propagate your choices through the rest of the installation process.

Below are the configurations you should enter, assuming you used the default network settings from the network firewall guide. If you did not, adjust these settings accordingly.

- *Application Server*:
- **Subnet:** 10.20.2.0/24
- **Address:** 10.20.2.2
- **Gateway:** 10.20.2.1
- **Name servers:** 8.8.8.8, 8.8.4.4
- **Search domains:** *should be left blank*
- *Monitor Server*:
- **Subnet:** 10.20.3.0/24
- **Address:** 10.20.3.2
- **Gateway:** 10.20.3.1
- **Name servers:** 8.8.8.8, 8.8.4.4
- **Search domains:** *should be left blank*

Select **Save** and press **Enter** to apply your settings. Then select **Done** and press **Enter**.

The default values on the **Configure Proxy** and **Configure Ubuntu archive mirror** screens should not need to be changed. Select **Done** for both.

Continue without updating the installer

With the network connection now active, the installer may alert you that a newer version of Ubuntu Sever is now available.

It is critical that you use the version of Ubuntu Server you downloaded and verified in the previous steps, rather than upgrading to the latest available version.

Select the **Continue without updating** option when prompted.

Full disk encryption - pros and cons

The use of Full Disk Encryption (FDE) with SecureDrop is **not recommended**. While FDE does offer data protection for devices that are powered down, SecureDrop's servers are designed to be always-on, with the exception of a nightly reboot after automatic upgrades are applied. Given this update schedule, with FDE enabled, the servers would become unreachable once every 24 hours until an administrator entered the full-disk encryption passphrase via the console, and during that time, *Sources* and *Journalists* would be unable to access your instance.

The increased responsibility for administrators, as well as the daily downtime and limited scenarios in which FDE would be a net security benefit, inform this recommendation, but you may make a decision based on your own requirements. (See this GitHub issue for more information.)

Setting up storage

On the **Guided storage configuration** screen, verify that **Use an entire disk** is checked, and that the server's local disk is selected. Also verify that **Set up this disk as an LVM group** is selected.

If you decided to set up FDE, despite the implications for administration overhead, select **Encrypt the LVM group with LUKS**, and enter and confirm the disk passphrase. Store this passphrase securely, as it will be required to unlock storage on every reboot.

Select **Done** and press **Enter** to move to the **Storage Configuration** screen. Review the configuration and select **Done** and press **Enter** to continue. Then, choose **Continue** on the **Confirm destructive action** dialog.

Configure account and hostname

On the **Profile setup** screen, configure the server's hostname and the administration account. The administrator account username and password should be the same for both servers:

- **Your name:** Specify the administrator account name, e.g. `SecureDrop Admin`
- **Your server's name:** Use `app` for the *Application Server*, and `mon` for the *Monitor Server*
- **Pick a username:** Specify the administrator account username, e.g. `sdadmin`
- **Choose a password:** Specify a strong password for the administrator account. A Diceware-generated passphrase is recommended.
- **Confirm your password:** Enter the password chosen above.

Select **Done** and press **Enter** to proceed.

Warning

The username and password you choose must be the same on both the *Application Server* and the *Monitor Server*. When you install SecureDrop on the servers from your *Admin Workstation* in a later step, you will only be allowed to enter one password, so it must be identical on both servers.

Decline upgrade to Ubuntu Pro

The SecureDrop servers should not be registered with Ubuntu Advantage. On the **Upgrade to Ubuntu Pro** screen, make sure **Skip for now** is selected, then choose **Continue**.

Set up SSH access

On the **SSH Setup** screen, enable **Install OpenSSH server**. Verify that **No** is selected for the **Import SSH Identity** option, as a custom SSH key will be created for the administration account later in the installation process.

Verify that **Allow password authentication over SSH** is selected, and choose **Done** to proceed.

Finish the installation

On the **Featured server snaps** screen, ensure that no snaps are selected and choose **Done** to start the server installation process.

Once the server installation is complete, choose **Reboot Now** to reboot the system.

Save the configurations

When you are done, make sure you save the following information:

- The IP address of the *Application Server*
- The IP address of the *Monitor Server*
- The non-root user's name and passphrase for the servers.

1.27 Install SecureDrop on the servers

Now that the servers are prepared, you are ready to install and configure the SecureDrop server on them. Like all future administrative tasks, this is performed from the `sd-admin` VM on the *Admin Workstation* you prepared earlier.

1.27.1 Test connectivity to servers

Having set up the firewall, you can plug the *Application Server* and the *Monitor Server* into the firewall. Your *Admin Workstation* should also be connected to the firewall.

If you are using a setup where there is a switch on the LAN port, plug the *Application Server* into the switch and plug the *Monitor Server* into the OPT1 port.

You should make sure you can connect from the *Admin Workstation* to both of the servers before continuing with the installation.

Open a terminal in `sd-admin` and verify that you can SSH into both servers, authenticating with your server administrator username (e.g. `sdadmin`) and password:

```
$ ssh <username>@<App IP address> hostname
app
$ ssh <username>@<Monitor IP address> hostname
mon
```

Tip

If you cannot connect, check the network firewall logs for clues.

1.27.2 Set up SSH keys

Ubuntu's default SSH configuration authenticates users with their passphrases; however, public key authentication is more secure, and once it's set up it is also easier to use. In this section, you will create a new SSH key for authenticating to both servers. Since the *Admin Workstation* was set up with SSH Client Persistence, this key will be saved on the *Admin Workstation* and can be used in the future to authenticate to the servers in order to perform administrative tasks.

First, generate the new SSH keypair:

```
ssh-keygen -t rsa -b 4096
```

You'll be asked to "Enter file in which to save the key" Type **Enter** to use the default location.

Given that this key is on the encrypted persistence of a Tails USB flash drive, you do not need to add an additional passphrase to protect the key. If you do elect to use a passphrase, note that you will need to manually type it (Tails' pinentry will not allow you to copy and paste a passphrase).

Once the key has finished generating, you need to copy the public key to both servers. Use `ssh-copy-id` to copy the public key to each server, authenticating with your passphrase:

```
ssh-copy-id <username>@<App IP address>
ssh-copy-id <username>@<Mon IP address>
```

Verify that you are able to authenticate to both servers by running the below commands. You should not be prompted for a passphrase (unless you chose to passphrase-protect the key you just created).

```
$ ssh <username>@<App IP address> hostname
app
$ ssh <username>@<Monitor IP address> hostname
mon
```

If you have successfully connected to the server via SSH, the terminal output will be name of the server to which you have connected ('app' or 'mon') as shown above.

1.27.3 Prepare configuration files

Make sure you have the following information and files ready before continuing:

- the *Application Server* local IP address
- the *Monitor Server* local IP address
- the *Submission Public Key* (*generated earlier*)
- the *Submission Key* fingerprint
- the email address that will receive alerts from OSSEC
- the GPG public key and fingerprint for the email address that will receive the alerts
- connection information for the SMTP relay that handles OSSEC alerts (see the *OSSEC Alerts Guide*)
- the username of a journalist who will be using SecureDrop (you can add more later)
- the username of the system admin

If configuring Daily Journalist Alert emails (this is optional and can be configured later), you will also need: - the *Journalist Alert Public Key* - the *Journalist Alert Public Key* fingerprint - the email address that will receive the Daily Journalist Alerts

1.27.4 Localization of the *Source Interface* and *Journalist Interface*

The *Source Interface* and *Journalist Interface* are translated in the following languages:

https://github.com/freedomofpress/securedrop/blob/develop/securedrop/i18n.rst

During the installation you will be given the opportunity to choose from a list of supported languages to display using the codes shown in parentheses.

Note

With a *Source Interface* displayed in French (for example), *Sources* submitting documents are likely to expect a *Journalist* fluent in French to be available to read the documents and follow up in that language.

1.27.5 OSSEC alerts public key

Before proceeding, you will need to copy the *OSSEC Alert Public Key* public key to `~/.config/securedrop-admin` in the `sd-admin` VM.

If you don't have your GPG key ready, you can run GnuPG on the command line in order to find, import, and export your public key. It's best to copy the key from a trusted and verified source, but you can also request it from keyservers using the known fingerprint. Looking it up by email address or a shorter key ID format could cause you to obtain a wrong, malicious, or expired key. Instead, we recommend you type out your fingerprint in groups of four (just like GPG prints it) enclosed by double quotes. The reason we suggest this formatting for the fingerprint is simply because it's easiest to type and verify correctly. In the code below simply replace `<fingerprint>` with your full, space-separated fingerprint:

Download your key and import it into the local keyring:

```
gpg --recv-key "<fingerprint>"
```

Note

It is important you type this out correctly. If you are not copy-pasting this command, we recommend you double-check you have entered it correctly before pressing enter.

Again, when passing the full public key fingerprint to the `--recv-key` command, GPG will implicitly verify that the fingerprint of the key received matches the argument passed.

Caution

If GPG warns you that the fingerprint of the key received does not match the one requested **do not** proceed with the installation. If this happens, please email us at securedrop@freedom.press.

Next we export the key to a local file.

```
gpg --export -a "<fingerprint>" > ossec.pub
```

Copy the key to a directory where it's accessible by the SecureDrop installation:

```
cp ossec.pub ~/.config/securedrop-admin/
```

The fingerprint is a unique identifier for an encryption (public) key. The short and long key ids correspond to the last 8 and 16 hexadecimal digits of the fingerprint, respectively, and are thus a subset of the fingerprint. The full fingerprint

must be the entire 40 hexadecimal digit GPG fingerprint for this same key, with all capital letters and no spaces. The following command will retrieve and format the fingerprint per our requirements:

```
gpg --with-colons --fingerprint "<fingerprint>" | grep "^fpr" | cut -d: -f10
```

The Postfix configuration enforces certificate verification, and requires both a valid certificate and STARTTLS support on the SMTP relay. By default the system CAs will be used for validating the relay certificate.

If you need to provide a custom CA to perform the validation, copy the cert file to `~/.config/securedrop-admin` add a new variable to `~/.config/securedrop-admin/site-specific`:

```
smtp_relay_cert_override_file: MyOrg.crt
```

where `MyOrg.crt` is the filename. The file will be copied to the server in `/etc/ssl/certs_local` and the system CAs will be ignored when validating the SMTP relay TLS certificate. Be sure to save `~/.config/securedrop-admin/site-specific` when you are finished.

1.27.6 Prepare SecureDrop server configuration

Open a terminal in `sd-admin` and run the following command, answering the prompts the the values that match your environment and SecureDrop installation:

```
securedrop-admin sdconfig
```

The script will automatically validate the answers you provided and display error messages if any problems are detected. The answers will be written to the file `~/.config/securedrop-admin/site-specific`.

Optional: configuring fingerprint verification

If you run your own mail server, you may wish to increase the security level used by Postfix for sending mail to `fingerprint`, rather than `secure`. Doing so will require an exact match for the fingerprint of TLS certificate on the SMTP relay. The advantage to fingerprint verification is additional security, but the disadvantage is potential maintenance cost if the fingerprint changes often. If you manage the mail server and handle the certificate rotation, you should update the SecureDrop configuration whenever the certificate changes, so that OSSEC alerts continue to send. Using fingerprint verification does not work well for popular mail relays such as smtp.gmail.com, as those fingerprints can change frequently, due to load balancing or other factors.

You can retrieve the fingerprint of your SMTP relay by running the command below (all on one line). Please note that you will need to replace `smtp.gmail.com` and `587` with the correct domain and port for your SMTP relay.

```
openssl s_client -connect smtp.gmail.com:587 -starttls smtp < /dev/null 2>/dev/null |
    openssl x509 -fingerprint -noout -in /dev/stdin | cut -d'=' -f2
```

If you are using Tails, you will not be able to connect directly with `openssl s_client` due to the default firewall rules. To get around this, proxy the requests over Tor by adding `torify` at the beginning of the command. The output of the command above should look like the following:

```
6D:87:EE:CB:D0:37:2F:88:B8:29:06:FB:35:F4:65:00:7F:FD:84:29
```

Finally, add a new variable to `~/.config/securedrop-admin/site-specific` as `smtp_relay_fingerprint`, like so:

```
smtp_relay_fingerprint: "6D:87:EE:CB:D0:37:2F:88:B8:29:06:FB:35:F4:65:00:7F:FD:84:29"
```

Specifying the fingerprint will configure Postfix to use it for verification on the next playbook run. (To disable fingerprint verification, simply delete the variable line you added, and rerun the playbooks.) Save `~/.config/securedrop-admin/site-specific` and exit the editor.

1.27.7 Install SecureDrop servers

Now you are ready to install! This process will configure the servers and install SecureDrop and all of its dependencies on the remote servers. In a terminal in `sd-admin` run the following command:

```
securedrop-admin install
```

You will be prompted to enter the sudo passphrase for the *Application Server* and *Monitor Server* (which should be the same).

The installation process will take some time. It will return you to the terminal prompt when complete.

If any errors occur while running the install, carefully inspect the error output. Considering saving any error messages for reference and troubleshooting.

Note

If you see an error running `securedrop-admin install`, and believe it may be an intermittent issue (for example, due to losing network connectivity to the servers), it is safe to run the `securedrop-admin install` command again. If you see the same issue consistently, then you will need to troubleshoot it.

If you see the error message "timeout (62s) waiting for privilege escalation prompt", try deleting the Ansible control path directory on your *Admin Workstation* (`rm -rf ~/.ansible/cp`) to reset the connection to the servers, then re-run the `securedrop-admin install`.

If you encounter other errors, we encourage you to submit a bug report, or to contact us at securedrop@freedom.press (GPG encrypted).

If needed, make edits to the file located at `~/.config/securedrop-admin/site-specific` as described *above*. If you continue to have issues, please submit a detailed issue notice on GitHub or send an email to securedrop@freedom.press.

Note

The SecureDrop install process configures a custom Linux kernel hardened with the grsecurity patch set. Only binary images are hosted in the apt repo. For source packages, see the Source Offer.

Once the installation is complete, addresses and credentials for each *Onion Service* will be available in the following files under `~/.config/securedrop-admin`:

1.27.8 V3 *Onion Services*

- `app-sourcev3-ths` contains the v3 onion address of the *Source Interface*.
- `app-journalist.auth_private` contains the onion address and private key providing access to the *Journalist Interface*.
- `app-ssh.auth_private` contains the onion address and private key providing SSH access to the *Application Server*.
- `mon-ssh.auth_private` contains the onion address and private key providing SSH access to the *Monitor Server*.
- `tor_v3_keys.json` contains the keypairs required for access to the *Journalist Interface* and SSH access to the servers - it is required for future runs of `securedrop-admin install`.

Warning

The three `.auth_private` files and the `tor_v3_keys.json` file contain secret keys that should not be shared with third parties, or copied from the *Admin Workstation* for any purpose other than tasks such as performing backups or onboarding new users.

The dynamic inventory file will automatically read the onion addresses from the `app-ssh.auth_private` and `mon-ssh.auth_private` files and use them to connect to the servers over SSH during subsequent playbook runs.

1.28 Apply configuration to *Admin Workstation*

With the servers installed and configured, the final step is to install the SecureDrop Inbox on the *Admin Workstation* and fully configure the machine.

1.28.1 Install and configure SecureDrop Inbox

- These steps should be performed from a `dom0` terminal. **Start a dom0 terminal** via  ▶  ▶ **Other Tools ▶ Xfce Terminal**.
- Configure infinite scrollback for your terminal via **Edit ▶ Preferences ▶ General ▶ Unlimited scrollback**. This helps to ensure that you will be able to review any error output printed to the terminal during the installation.
- Finally, in the `dom0` terminal, run the command:

```
sdw-admin --apply
```

This command will take a considerable amount of time and approximately 4GB of bandwidth, as it sets up multiple VMs and installs supporting packages. When the command finishes, reboot the machine to complete the installation. This SecureDrop Workstation is finally ready to use!

1.28.2 Test the *Admin Workstation*

The preflight updater will start automatically after logging into the system. Please follow the preflight updater's instructions.

Note

If you close SecureDrop Inbox during your session, you can launch it again using the SecureDrop icon on the desktop.

Once the update check is complete, the SecureDrop Client will launch. Log in using an existing journalist account and verify that *Sources* are listed and submissions can be downloaded, decrypted, and viewed.

1.28.3 Enable password copy and paste

If you use KeePassXC in the `vault` VM to manage login credentials, you can enable the user to copy passwords to SecureDrop Inbox using inter-VM copy and paste. While this is relatively safe, we recommend reviewing the section *Managing Clipboard Access* of this guide, which goes into further detail on the security considerations for inter-VM copy and paste.

The password manager runs in the networkless `vault` VM, and the SecureDrop Inbox application runs in the `sd-app` VM. To permit this one-directional clipboard use, issue the following command in `dom0`:

```
qvm-tags vault add sd-send-app-clipboard
```

Confirm that the tag was correctly applied using the `ls` subcommand:

```
qvm-tags vault ls
```

To revoke this configuration change later or correct a typo, you can use the `del` subcommand, e.g.:

```
qvm-tags vault del sd-send-app-clipboard
```

Troubleshooting `sdw-admin`

1.28.4 "Failed to return clean data"

An error similar to the following may be displayed during an installation or update:

```
sd-log:
      ----------
      _error:
          Failed to return clean data
      retcode:
          None
      stderr:
      stdout:
          deploy
```

This is a transient error that may affect any of the SecureDrop Workstation VMs. To clear it, run the installation command or update again.

1.28.5 "Temporary failure resolving"

Transient network issues may cause an installation to fail. To work around this, verify that you have a working Internet connection, and re-run the `sdw-admin --apply` command.

1.29 Create an admin account on the *Journalist Interface*

In order for any user (admin or *Journalist*) to access the *Journalist Interface*, they need:

1. The `auth-cookie` for the *Journalist Interface*'s ATHS
2. An account on the *Journalist Interface*, which requires the following credentials to log in:
    - Username
    - Passphrase
    - *Two-Factor Authentication* code

You should create a separate account on the *Journalist Interface* for each user who needs access. This makes it easy to enable or disable access to the *Journalist Interface* on an individual basis, so you can grant access to new users or revoke access for users who have left the organization or should no longer be allowed to access the *Journalist Interface*.

There are two types of accounts on the *Journalist Interface*: admin accounts and normal accounts. Admins accounts are like normal accounts, but they are additionally allowed to manage (add, change, delete) other user accounts through the web interface.

You must create the first admin account on the *Journalist Interface* by running a command on the *Application Server*. After that, the admin can create additional accounts through the web-based *Journalist Interface*.

To create an admin account via the command line, *SSH to the *Application Server**, then:

```
sudo -u www-data bash
cd /var/www/securedrop
./manage.py add-admin
```

Follow the prompts.

A secure diceware passphrase will be generated by *manage.py*. You will see output like this:

```
This *Journalist*'s passphrase is: delivery propose requisite stunner dragonfly␣
↪unstamped stowaway
```

Passphrases include the spaces between the words, but not leading or trailing whitespace. Be sure to save this passphrase in the appropriate KeePassXC database.

Once that's done, you should open Tor Browser and navigate to the *Journalist Interface*'s onion address. Verify that you can log in to the *Journalist Interface* with the admin account you just created.

For adding more user accounts, please refer now to our *Admin Interface Guide*.

Note

You can now set a custom logo image on your web interfaces by following the *Updating the logo image* documentation.

1.30 Test the installation

1.30.1 Test Connectivity

SSH to both servers over Tor

Assuming you haven't disabled SSH over Tor, SSH access will be restricted to the Tor network.

On the *Admin Workstation*, you should be able to SSH to the *Application Server* and the *Monitor Server* from the `sd_admin` VM

```
ssh app
ssh mon
```

The SSH aliases should have been configured automatically by running the `securedrop-admin localconfig` command. If you're unable to connect via aliases, try using the verbose command format to troubleshoot:

```
ssh <username>@<app .onion>
ssh <username>@<mon .onion>
```

Tip

Check the `app-ssh.auth_private` and `mon-ssh.auth_private` files in the `~/.config/securedrop-admin` directory in the `sd_admin` VM to find the ssh onion service addresses. The files contain one line with 4 colon-delimited fields. The address is the first 56-character field, just add a `.onion` at the end.

Log in to both servers via TTY

All access to the SecureDrop servers should be performed over SSH from the *Admin Workstation*. To aid in troubleshooting, login via a physical keyboard attached to the server is also supported.

1.30.2 Sanity-check the installation

On each server:

1. Check that you can execute privileged commands by running `sudo su`.
2. Verify that you are booted into a grsec kernel: run `uname -r` and verify that the name of the running kernel ends with `-grsec`.
3. Check the current applied iptables rules with `iptables-save`. It should output *approximately* 50 lines.
4. You should have received an email alert from OSSEC when it first started. If not, review our *OSSEC Alerts Guide*.

On the *Application Server*:

1. Check the AppArmor status with `sudo aa-status`. On a production instance all profiles should be in `enforce` mode.

1.30.3 Test the web interfaces

1. Make sure the *Source Interface* is available, and that you can make a submission.
   - Open the *Source Interface* in Tor Browser by clicking on its desktop shortcut. Proceed through the codename generation (copy this down somewhere) and submit a test message or file.
   - Usage of the *Source Interface* is covered by our *Source User Manual*.
2. Test that you can access the *Journalist Interface*, and that you can log in as the admin user you just created.
   - Open the *Journalist Interface* in Tor Browser by clicking on its desktop shortcut. Enter your passphrase and two-factor code to log in.
   - If you have problems logging in to the *Journalist Interface*, SSH to the *Application Server* and restart the time synchronization daemon to synchronize the time: `sudo systemctl restart systemd-timesyncd`. Also check that your smartphone's time is accurate and set to network time in its device settings.
3. Test replying to the test submission.
   - While logged in as an administrator, you can send a reply to the test source submission you made earlier.
   - Usage of the *Journalist Interface* is covered by our *Journalist User Manual*.
4. Verify that the test source account received the reply.
   - Within Tor Browser, navigate back to the *Source Interface* and use your previous test source codename to log in (or reload the page if it's still open) and check that the reply you just made is present.
5. Remove the test submissions you made prior to putting SecureDrop to real use. On the main *Journalist Interface* page, select all *Sources* and click **Delete selected**.

Once you've tested the installation and verified that everything is working, see *How to Use SecureDrop*.

1.31 Provisioning USB *Export Devices*

The *Journalist Workstation* supports the export of submissions from the SecureDrop Inbox to a LUKS- or VeraCrypt-encrypted USB flash drive, referred to as an *Export Device*.

1.31.1 Creating a LUKS-encrypted drive

Note

LUKS-encrypted drives can only be used with Linux-based systems such as Tails. For compatibility with macOS and Windows systems, use VeraCrypt.

In order to provision a LUKS-encrypted *Export Device* for use a *Journalist Workstation*, you will need a fresh USB flash drive and a SecureDrop Workstation.

- First, boot the SecureDrop Workstation.
- Next, open the Disks utility: **Applications ▶ Utilities ▶ Disks**.
- Connect the fresh USB flash drive and select it in the list in the left-hand panel.

Warning

The formatting operation will wipe any data on an existing partition. Make sure that you select the correct device!

- Click the interlocking gear icon under the drive volumes schematic in the right-hand panel and choose **Format Partition…**.
- Select the following options in the Format Volume dialog:
  - Volume Name: Transfer
  - Type: Ext4, with the "Password protect volume (LUKS)" option enabled
- Then, click **Next**. You will be prompted to set a password. This password should be strong - a 6-word Diceware passphrase is highly recommended.
- Once the password is set, click **Format**, then when prompted, click **Format** again. The formatting process should take only a few seconds.
- Once formatting is complete, you will need to provide the *Export Device* and its decryption password to the *Journalist Workstation* users. Make sure that they store it and its password securely, as it will contain decrypted submissions.

1.31.2 Creating a VeraCrypt-encrypted drive

Warning

If you plan to use your *Export Device* with computers running macOS 15 ("Sequoia") or later, you must also perform the VeraCrypt setup on that version of macOS.

- If it isn't already done, download and install the VeraCrypt software.
- Start VeraCrypt from your computer's application or software interface.
- Click **Create Volume**

- Select **Encrypt a non-system partition/drive** and click **Next**.
- Select **Standard VeraCrypt volume** and click **Next**
- Connect your fresh USB flash drive and click **Select Device…** to select it.
  - You may see a warning that says "We strongly recommend that inexperienced users create a VeraCrypt file container on the selected device/partition, instead of attempting to encrypt the entire device/partition." We disagree with this recommendation, so click **Yes**.
  - Click **Next** to advance.

Warning

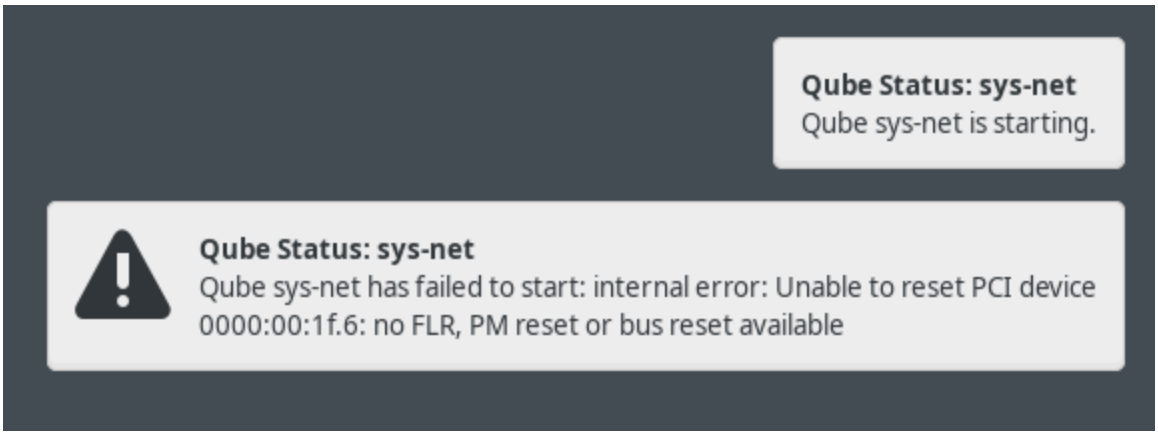
The formatting operation will wipe any data on an existing partition. Make sure that you select the correct device!

- You will be prompted to set a password. This password should be strong - a 6-word Diceware passphrase is highly recommended.
- You will be asked if you need to store large files, select **No** and click **Next**.
- Select the following options in the Volume Format dialog:
  - Filesystem: exFAT
  - Quick Format: unselected
- Click **Next**. VeraCrypt will now collect entropy from your mouse movements. Randomly move your mouse cursor around the screen until the progress bar is filled up. Then click **Format**.
  - You will be reminded that all files on the device will be erased and lost and given a final confirmation to begin. Click **Yes**.
- Wait until VeraCrypt says "The VeraCrypt volume has been successfully created." Until this pops up, it may look like the program is frozen, but it's running in the background.
- Click **OK** and then **Exit** to finish formatting process.
- Once formatting is complete, you will need to provide the *Export Device* and its decryption password to the *Journalist Workstation* users. Make sure that they store it and its password securely, as it will contain decrypted submissions.

1.32 Troubleshooting Qubes issues during installation

1.32.1 "Unable to reset PCI device"

On some hardware, network devices (Ethernet and Wi-Fi) will not immediately work out of the box and require a one-time manual configuration on install. After Qubes starts for the first time, `sys-net` will fail to start:



Open a `dom0` terminal via ▶ ▶ **Other Tools** ▶ **Xfce Terminal**, and run the following command to list the devices connected to the `sys-net` VM.

```
qvm-pci ls sys-net
```

This will return the two devices (Ethernet and WiFi) that are connected to `sys-net`:

```
BACKEND:DEVID  DESCRIPTION                                                    ␣
↪USED BY
dom0:00_14.3   Network controller: Intel Corporation                          ␣
↪sys-net
dom0:00_1f.6   Ethernet controller: Intel Corporation Ethernet Connection (5) I219-V ␣
↪sys-net
```

For both device IDs (e.g. `dom0:00_1f.6` and `dom0:00_14.3`), you will need to detach and re-attach the device to `sys-net`, then restart `sys-net` as follows:

```
qvm-pci detach sys-net dom0:00_14.3
qvm-pci detach sys-net dom0:00_1f.6
qvm-pci attach sys-net --persistent --option no-strict-reset=True dom0:00_14.3
qvm-pci attach sys-net --persistent --option no-strict-reset=True dom0:00_1f.6
qvm-start sys-net
```

`sys-net` should now start, and network devices will be functional. This change is only required once on first install. See the Qubes documentation of this issue for more information.

1.32.2 Full system freezes

A known issue with some hardware results in Qubes fully freezing. If you encounter this issue, you will need to forcibly restart your computer, usually by holding down the power button.

When you boot up, you will see a black-and-white menu with the following options:

```
Qubes, with Xen hypervisor
Advanced options for Qubes (with Xen hypervisor)
UEFI Firmware Settings
```

While `Qubes, with Xen hypervisor` is selected, press `e` to edit the option. You should now see a rudimentary edit interface.

Find the line that starts with `multiboot2 /xen-` and ends with `${xen_rm_opts}`. Use the arrow keys to move your cursor to before `${xen_rm_opts}` and type `cpufreq=xen:hwp=off` (leave a space between `off` and the `$`.

Press `Ctrl-x` to continue with booting. This will fix the current boot, we now need to make the fix permanent.

Once Qubes has started and you have logged in, open a `dom0` terminal via  ▶  ▶ **Other Tools** ▶ **Xfce Terminal** and type `sudo nano /etc/default/grub` to start an editor.

Move your cursor to the bottom of the file and add: `GRUB_CMDLINE_XEN_DEFAULT="$GRUB_CMDLINE_XEN_DEFAULT cpufreq=xen:hwp=off"`

Press `Ctrl-x`, then `y`, and then `Enter` to save the file.

Finally, in the terminal run `sudo grub2-mkconfig -o /boot/grub2/grub.cfg`. The workaround will now automatically be applied going forwards.

1.33 Troubleshooting OSSEC

Some OSSEC alerts should begin to arrive as soon as the installation has finished.

The easiest way to test that OSSEC is working is to SSH to the Monitor Server and run `systemctl restart ossec`. This will trigger an Alert level 3 saying: "Ossec server started."

So you've finished installing SecureDrop, but you haven't received any OSSEC alerts. First, check your spam/junk folder. If they're not in there, then most likely there is a problem with the email configuration. In order to find out what's wrong, you'll have to SSH to the Monitor Server and take a look at the logs. To examine the mail log created by Postfix, run the following command:

```
tail /var/log/mail.log
```

The output will show you attempts to send the alerts and provide hints as to what went wrong. Here's a few possibilities and how to fix them:

Problem	Solution
Connection timed out	Check that the hostname and port is correct in the relayhost line of `/etc/postfix/main.cf`
Server certificate not verified	Check that the relay certificate is valid (for more detailed help, see *Troubleshooting SMTP TLS*). Consider adding `smtp_relay_cert_override_file` to `prod_specific.yml` as described above.
Authentication failure	Edit `/etc/postfix/sasl_passwd` and make sure the username, domain and password are correct. Run `postmap /etc/postfix/sasl_passwd` to update when finished.

After making changes to the Postfix configuration, you should run `systemctl reload postfix` and test the new settings by restarting the OSSEC service.

Tip

If you change the SMTP relay port after installation for any reason, you must update the SMTP relay port using `securedrop-admin sdconfig` and deploy using `securedrop-admin install`.

1.33.1 Useful log files for OSSEC

Other log files that may contain useful information:

/var/log/procmail.log
Includes lines for sending mail containing OSSEC alerts.

/var/log/syslog
Messages related to grsecurity, AppArmor and iptables.

/var/ossec/logs/ossec.log
OSSEC's general operation is covered here.

/var/ossec/logs/alerts/alerts.log
Contains details of every recent OSSEC alert.

Tip

Remember to encrypt any log files before sending via email, for example to securedrop@freedom.press, in order to protect security-related information about your organization's SecureDrop instance.

1.33.2 Not receiving emails

Some mail servers require that the sending email address match the account that authenticated to send mail. By default the *Monitor Server* will use `ossec@ossec.server` for the from line, but your mail provider may not support the mismatch between the domain of that value and your real mail host. If the Admin email address (configured as `ossec_alert_email` in `~/.config/securedrop-admin/site-specific`) does not start receiving OSSEC alerts updates shortly after the first playbook run, try setting `ossec_from_address` in `~/.config/securedrop-admin/site-specific` to the full email address used for sending the alerts, then run the playbook again.

1.33.3 Message failed to encrypt

If OSSEC cannot encrypt the alert to the *OSSEC Alert Public Key* for the Admin email address (configured as `ossec_alert_email` in `~/.config/securedrop-admin/site-specific`), the system will send a static message instead of the scheduled alert:

> Failed to encrypt OSSEC alert. Investigate the mailing configuration on the Monitor Server.

Check the GPG configuration vars in `~/.config/securedrop-admin/site-specific`. In particular, make sure the GPG fingerprint matches that of the public key file you exported.

1.33.4 Troubleshooting SMTP TLS

Your choice of SMTP relay server must support STARTTLS and have a valid server certificate. By default, the *Monitor Server*'s Postfix configuration will try to validate the server certificate using the default root store (in Ubuntu, this is maintained in the `ca-certificates` package). You can override this by setting `smtp_relay_cert_override_file` as described earlier in this document.

In either situation, it can be helpful to use the `openssl` command line tool to verify that you can successfully connect to your chosen SMTP relay securely. We recommend doing this before running the playbook, but it can also be useful as part of troubleshooting OSSEC email send failures.

In either case, start by attempting to make a STARTTLS connection to your chosen `smtp_relay:smtp_relay_port` (get the values from your `group_vars/all/site-specific` file). On a machine running Ubuntu, run the following `openssl` command, replacing `smtp_relay` and `smtp_relay_port` with your specific values:

```
openssl s_client -showcerts -starttls smtp -connect smtp_relay:smtp_relay_port < /dev/
↪null 2> /dev/null
```

Note that you will not be able to run this command on the Application Server because of the firewall rules. You can run it on the Monitor Server, but you will need to run it as the Postfix user (again, due to the firewall rules):

```
sudo -u postfix openssl s_client -showcerts -starttls smtp -connect smtp.gmail.com:587 <␣
↪/dev/null 2> /dev/null
```

If the command fails with "Could not connect" or a similar message, then this mail server does not support STARTTLS. Verify that the values you are using for `smtp_relay` and `smtp_relay_port` are correct. If they are, you should contact the admin of that relay and talk to them about supporting STARTTLS, or consider using another relay that already has support.

If the command succeeds, the first line of the output should be "CONNECTED" followed by a lot of diagnostic information about the connection. You should look for the line that starts with "Verify return code", which is usually one of the last lines of the output. Since we did not give `openssl` any information about how to verify certificates in the previous command, it should be a non-zero value (indicating verification failed). In my case, it is `Verify return code: 20 (unable to get local issuer certificate)`, which indicates that openssl does not know how to build the certificate chain to a trusted root.

If you are using the default verification setup, you can check whether your cert is verifiable by the default root store with `-CApath`:

```
openssl s_client -CApath /etc/ssl/certs -showcerts -starttls smtp -connect smtp_
↪relay:smtp_relay_port < /dev/null 2> /dev/null
```

For example, if I'm testing Gmail as my SMTP relay (`smtp.gmail.com:587`), running the `openssl` with the default root store results in `Verify return code: 0 (ok)` because their certificate is valid and signed by one of the roots in the default store. This indicates that can be successfully used to securely relay email in the default configuration of the *Monitor Server*.

If your SMTP relay server does not successfully verify, you should use the return code and its text description to help you diagnose the cause. Your cert may be expired, in which case you should renew it. It may not be signed by a trusted CA, in which case you should obtain a signature from a trusted CA and install it on the mail server. It may not have the right hostnames in the Common Name or Subject Alternative Names, in which case you will need to generate a new CSR with the correct hostnames and then obtain a new certificate and install it. Etc., etc.

If you are *not* using the default verification setup, and intentionally do not want to use a certificate signed by one of the default CA's in Ubuntu, you can still use `openssl` to test whether you can successfully negotiate a secure connection. Begin by copying your certificate file (`smtp_relay_cert_override_file` from `group_vars/all/site-specific`) to the computer you are using for testing. You can use `-CAfile` to test if your connection will succeed using your custom root certificate:

```
openssl s_client -CAfile /path/to/smtp_relay_cert_override_file -showcerts -starttls␣
↪smtp -connect smtp_relay:smtp_relay_port < /dev/null 2> /dev/null
```

Finally, if you have a specific server in mind but are not sure what certificate you need to verify the connection, you can use the output of `openssl s_client` to figure it out. Since we have `-showcerts` turned on, `openssl` prints

the entire certificate chain it receives from the server. A properly configured server will provide all of the certificates in the chain up to the root cert, which needs to be identified as "trusted" for the verification to succeed. To see the chain, find the part of the output that start with `Certificate chain`. It will look something like this (example from `smtp.gmail.com`, with certificate contents snipped for brevity):

```
---
Certificate chain
0 s:/C=US/ST=California/L=Mountain View/O=Google Inc/CN=smtp.gmail.com
i:/C=US/O=Google Inc/CN=Google Internet Authority G2
-----BEGIN CERTIFICATE-----
<snip>
-----END CERTIFICATE-----
1 s:/C=US/O=Google Inc/CN=Google Internet Authority G2
i:/C=US/O=GeoTrust Inc./CN=GeoTrust Global CA
-----BEGIN CERTIFICATE-----
<snip>
-----END CERTIFICATE-----
2 s:/C=US/O=GeoTrust Inc./CN=GeoTrust Global CA
i:/C=US/O=Equifax/OU=Equifax Secure Certificate Authority
-----BEGIN CERTIFICATE-----
<snip>
-----END CERTIFICATE-----
---
```

The certificates are in reverse order from leaf to root. `openssl` handily prints the Subject (`s:`) and Issuer (`i:`) information for each cert. In order to find the root certificate, look at the Issuer of the last certificate. In this case, that's `Equifax Secure Certificate Authority`. This is the root certificate that issued the first certificate in the chain, and it is what you need to tell Postfix to use in order to trust the whole connection.

Actually obtaining this certificate and establishing trust in it is beyond the scope of this document. Typically, if you are using your own SMTP relay with a custom CA, you will be able to obtain this certificate from an intranet portal or someone on your IT staff. For a well-known global CA, you can obtain it from the CA's website. For example, a quick search for "Equifax Secure Certificate Authority" finds the web page of GeoTrust's Root Certificates, which have accompanying background information and are available for download.

Once you have the root certificate file, you can use `-CAfile` to test that it will successfully verify the connection.

1.34 Migration overview

1.35 Migrating from a Tails-based SecureDrop

1.35.1 Pre-install tasks:

1. Apply BIOS updates and check settings
2. Download and verify Qubes OS
3. Install Qubes OS
4. (Hardware-dependent) Apply USB fixes
5. Apply updates to system templates

1.35.2 Install tasks:

1. Copy the *Submission Key*
2. Copy *Journalist Interface* details
3. Copy SecureDrop login credentials
4. Download and install SecureDrop Workstation
5. Configure SecureDrop Workstation
6. Test the Workstation

Prerequisites

In order to install SecureDrop Workstation and configure it to use an existing SecureDrop instance, you will need the following:

- A Qubes-compatible laptop based on the *hardware* recommendations.
- Qubes installation medium - this guide assumes the use of a USB 3.0 flash drive. Qubes may also be installed via optical media, which may make more sense depending on your security concerns.

  Note

  A USB flash drive with a Type-A connector is recommended, as USB-C ports may be disabled on your computer when the BIOS settings detailed below are applied.

- A working computer (Linux is recommended and assumed in this guide) to use for verification and creation of the Qubes installation medium.

  Note

  Tails can be used to perform the tasks below, but due to the size of the Qubes installation ISO, it may make sense to download it on another computer rather than via Tor, and then to use a USB flash drive to transfer it to Tails for verification and creation of the installation medium.

- A password manager or other system to generate and store strong passphrases for Qubes full disk encryption (FDE) and user accounts.

A basic knowledge of the Qubes OS is helpful.

Pre-install tasks

1.35.3 Apply BIOS updates and check settings

Before beginning the Qubes installation, make sure that your Qubes-compatible computer's BIOS is updated to the latest available version. For more details about this process, see the section on *Automatic BIOS updates*.

Once the BIOS is up-to-date, boot into the BIOS setup utility and update its settings. Note that not all BIOS versions will support the items listed, but if available following changes are recommended:

- Ensure the internal clock is correct.
- Set a password to access the BIOS (and record the password in your password manager).
- Disable BIOS downgrades.
- Enable Data Execution Prevention.

- Enable virtualization support (required for Qubes OS). - for Intel-based devices, **Intel VT-d** and **Intel VT-x** should be enabled - for AMD-based devices, **AMD-VI** and **AMD-V** should be enabled
- Disable unnecessary I/O options such as Wireless WAN and Bluetooth.
- Disable unnecessary network options such as Wake-on-LAN and UEFI network stacks.
- Disable Thunderbolt ports, or any other ports that allow Direct Memory Access (DMA).
- Enable any physical tamper detection options.
- Disable Computrace.
- Disable SecureBoot.

If the Qubes hardware compatibility list entry for your computer recommends the use of Legacy Mode for boot, change that setting in the BIOS as well.

1.35.4 Disable SecureBoot

SecureBoot is a feature available on most systems that, when enabled, does not allow any operating system to boot that has not been signed by a trusted key. By only booting to operating systems that are properly signed, you can be sure that the OS itself has not been corrupted or tampered with, at least at the boot level.

SecureBoot must be disabled on the server and Workstation hardware. SecureDrop installs a hardened, security-focused version of the Linux kernel (grsec) that does not support SecureBoot. If SecureBoot is enabled on either of the servers during the install, you will receive a pre-install error reminding you that it must be turned off before the installation can proceed.

Likewise, SecureBoot is not fully supported by QubesOS, and cannot be used with *SecureDrop Workstations*.

For instructions on how to enable or disable the SecureBoot feature for your device, please consult the manufacturer's manual for BIOS settings, as they differ for each make and model.

1.35.5 Download and verify Qubes OS

On the working computer, download the Qubes OS ISO and cryptographic hash values for version `4.2.4` from https://www.qubes-os.org/downloads/. The ISO is 6.8 GB approximately, and may take some time to download based on the speed of your Internet connection.

Follow the linked instructions to verify the ISO. Ensure that the ISO and hash values are in the same directory, then run:

```
gpg --keyserver-options no-self-sigs-only,no-import-clean --fetch-keys https://keys.
↪qubes-os.org/keys/qubes-release-4.2-signing-key.asc
gpg -v --verify Qubes-R4.2.4-x86_64.iso.DIGESTS
sha256sum -c Qubes-R4.2.4-x86_64.iso.DIGESTS
```

The output should look like this:

```
gpg: requesting key from 'https://keys.qubes-os.org/keys/qubes-release-4.2-signing-key.
↪asc'
gpg: key E022E58F8E34D89F: public key "Qubes OS Release 4.2 Signing Key" imported
gpg: Total number processed: 1
gpg:               imported: 1
gpg: no ultimately trusted keys found

gpg: armor header: Hash: SHA256
gpg: original file name=''
```

(continues on next page)

(continued from previous page)

```
gpg: Signature made Mon 17 Feb 2025 12:00:00 AM EST
gpg:                using RSA key 9C884DF3F81064A569A4A9FAE022E58F8E34D89F
gpg: using pgp trust model
gpg: Good signature from "Qubes OS Release 4.2 Signing Key" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Primary key fingerprint: 9C88 4DF3 F810 64A5 69A4  A9FA E022 E58F 8E34 D89F
gpg: textmode signature, digest algorithm SHA256, key algorithm rsa4096
Qubes-R4.2.4-x86_64.iso: OK
sha256sum: WARNING: 20 lines are improperly formatted
```

Specifically, you will want to make sure that you see "Good signature" listed in the text. If it does not report a good signature, try deleting the ISO and downloading it again.

Once you've verified the ISO, copy it to your installation medium - for example, if using Linux and a USB flash drive, using the command:

```
sudo dd if=Qubes-R4.2.4-x86_64.iso of=/dev/sdX bs=1048576 && sync
```

where `if` is set to the path to your downloaded ISO file and `of` is set to the block device corresponding to your USB flash drive. Note that any data on the USB flash drive will be overwritten.

Caution

Make sure to verify that you have the correct device name using, for example, the `lsblk` command. You should write to the full device (eg. `/dev/sdc`) rather than to a partition (eg. `/dev/sdc1`).

1.35.6 Install Qubes OS (estimated wait time: 30-45 minutes)

Before starting the installation, please ensure that:

- the computer is charging
- all USB devices like YubiKeys, mice and keyboards are disconnected

To begin the Qubes installation, connect the Qubes installation drive you just created to your target computer and boot from it. You may need to bring up a boot menu at startup to do so - on Lenovo laptops, for example, you can do so by pressing **F12** on boot.

Follow the installation documentation to install Qubes on your computer, ensuring that you:

- Use English - United States as the setup language. (This requirement will be dropped in a future version).
- Use all available storage space for the installation (as the computer should be dedicated to SecureDrop Workstation).
- Set a strong full disk encryption (FDE) passphrase - a 6-word Diceware passphrase is recommended.
- Create an administrative account named `user` with a strong password.

Note

Qubes is not intended to have multiple user accounts, so your account name and password will be shared by all SecureDrop Workstation users. The password will be required to log in and unlock the screen during sessions - choosing something strong but memorable and easily typed is recommended!

Once the installation is complete, you will be prompted to reboot into Qubes. Reboot, removing the install USB flash drive when the computer restarts.

You will be prompted to enter the FDE passphrase set during installation.

After the disk is unlocked and Qubes starts, you will be prompted to complete the initial setup. Click the Qubes OS icon.

On the configuration screen, ensure that the following options are checked:

- Default Template should be set to "Fedora 41 Xfce"
- "Create default system qubes (sys-net, sys-firewall, default DispVM)"
- "Make sys-firewall and sys-usb disposable"

If there is a grayed out option "USB qube configuration disabled", make a note of this. An additional setup step will be required (see next section).

Finally, click **Finish Configuration** to set up the default system TemplateVMs and AppVMs.

Once the initial setup is complete, the login dialog will be displayed. Log in using the username and password set during installation.

1.35.7 (Hardware-dependent) Apply USB fixes

If, during the installation, you encountered the grayed out option "USB qube configuration disabled", you must now create a VM to access your USB devices. If you did not encounter this issue, you can skip this section.

To create a USB qube, open a `dom0` terminal via  ▶  ▶ **Other Tools** ▶ **Xfce Terminal**.

Tip

For quicker access, you can add the `dom0` terminal to the "Favorites" section of the Qubes menu (identified by a bookmark symbol). Right-click the entry and select **Add to favorites**. To remove it at a later time, right-click the entry in your list of favorites and select **Remove from favorites**.

Run the following command:

```
sudo qubesctl state.sls qvm.sys-usb
```

After the command exits, confirm that you see an entry "Service: sys-usb" in the Qubes menu. If `sys-usb` is not running, you can start it with the command `qvm-start sys-usb` in `dom0`. Once `sys-usb` is running, click the devices widget in the upper right panel to expand a listing of all devices detected by Qubes OS.

Now, insert a safe USB device you intend to use with the SecureDrop Workstation. Click the devices widget again. Does the newly attached USB device appear in the list? If so, USB support is working and you can proceed with the installation. If you do encounter the error message "Denied qubes.InputKeyboard from sys-usb to dom0", you need to additionally enable USB keyboard support:

```
sudo qubesctl state.sls qvm.usb-keyboard
```

While we recommend against the use of a USB keyboard for security reasons, this error can also occur in combination with other USB devices on some hardware.

1.35.8 Apply dom0 updates (estimated wait time: 15-30 minutes)

dom0 is the most trusted domain on Qubes OS, and has privileged access to all other VMs. As such, it is important to ensure that all available security updates have been applied to dom0 as the first step after the installation.

After logging in, use the network manager widget in the upper-right panel to configure your network connection.

Open a dom0 terminal from the Qubes Application menu (the  icon in the upper left corner) by selecting  ▶  **(left-hand side) ▶ Other Tools ▶ Xfce Terminal**. Run the following command:

```
sudo qubes-dom0-update -y
```

Wait for all updates to complete. If you encounter an error during this stage, please contact us for assistance, as it may not be safe to proceed with the installation.

After updating dom0, reboot the workstation to ensure that all updates have taken effect for your active session.

1.35.9 Apply updates to system templates (estimated wait time: 45-60 minutes)

After logging in again, confirm that the network manager successfully connects you to the configured network. If necessary, verify the network settings using the network manager widget.

- Next, configure Tor via  ▶ **Service ▶ sys-whonix ▶ Anon Connection Wizard**. In most cases, choosing the default **Connect** option is best. Click **Next**, then **Next** again. Then, if Tor connects successfully, click **Finish**. If Tor fails to connect, make sure your network conection is up and does not filter Tor connections, then try again.

  Note

  If Tor connections are blocked on your network, you may need to configure Tor to use bridges in order to get a connection. For more information, see the Anon Connection Wizard documentation.

- Once Tor has connected, launch the Qubes Update tool via  ▶ **Qubes Tools ▶ Qubes Update** to update the system VMs. in the `[Dom0] Qubes Update` window, check all entries in the list above except for dom0 (which you have already updated in the previous step). Then, click **Update**. The system's VMs will be updated sequentially - this may take some time. When the updates are complete, click **Next**. You will then be prompted to **Finish and restart/shutdown 4 qubes.** Go ahead and do so, and allow time for them to restart.

Installing SecureDrop Workstation

1.35.10 Download SecureDrop Workstation packages

First, you must configure the Qubes-Contrib repo, then download the SecureDrop Workstation packages.

- Make sure that network connection is enabled using the network manager widget in the upper right panel.
- Next, in a dom0 terminal ( ▶  ▶ **Other ▶ Xfce Terminal**):

```
sudo qubes-dom0-update -y qubes-repo-contrib
sudo qubes-dom0-update --clean -y securedrop-workstation-keyring
```

- The SecureDrop Relase keyring will be installed on your machine. Wait 15 seconds for the key to be imported into the rpm database. Then:

```
sudo qubes-dom0-update --clean -y securedrop-workstation-dom0-config
sudo dnf -y remove qubes-repo-contrib
```

1.35.11 Import KeePassXC database

If you have a KeePassXC database on your Tails-based *Admin Workstation* USB flash drive, you should copy it to the `vault` VM on the new Qubes-based *Admin-Workstation*.

Qubes OS comes with the KeePassXC password manager preinstalled in the `vault` VM.

- Open the KeePassXC program in the *vault* VM
- Select **Database ▶ Open database**, and navigate to the location of **/path/to/Passwords.kdbx**, select it, and click **Open**
- Leave the password blank and click **OK**. If you receive an "Unlock failed" prompt, click **Retry with empty password**.
- Edit entries as required.
- Select **Database ▶ Save Database** to save your changes.

The next time you use KeepassXC in *vault*, the database at `/path/to/Passwords.kdbx` will be selected by default.

KeePassXC will show a warning every time you attempt to open a database without entering a password. Because your persistent volume is encrypted, setting up this additional password is not strictly required. It provides some additional protection, e.g., if a computer is left running, at the cost of convenience.

For passwordless access without warnings, you can protect the database using a key file, via **Database ▶ Database settings ▶ Security ▶ Add additional protection ▶ Add Key File ▶ Generate**. This key file has to be stored in your Persistent folder and it must be selected when you open the database.

After configuring the password database, restart KeePassXC once to verify that you are able to access it as expected.

Warning

You will not be able to access your passwords if you forget the full disk encryption or the location of the key file used to protect the database.

1.35.12 Configure SecureDrop Workstation

Now that your new Qubes-based *Admin-Workstation* is prepared, you can proceed with importing the correct SecureDrop server details and *Submission Private Key* from your Tails-based *Journalist Workstation* and *Secure Viewing Station* USB flash drives.

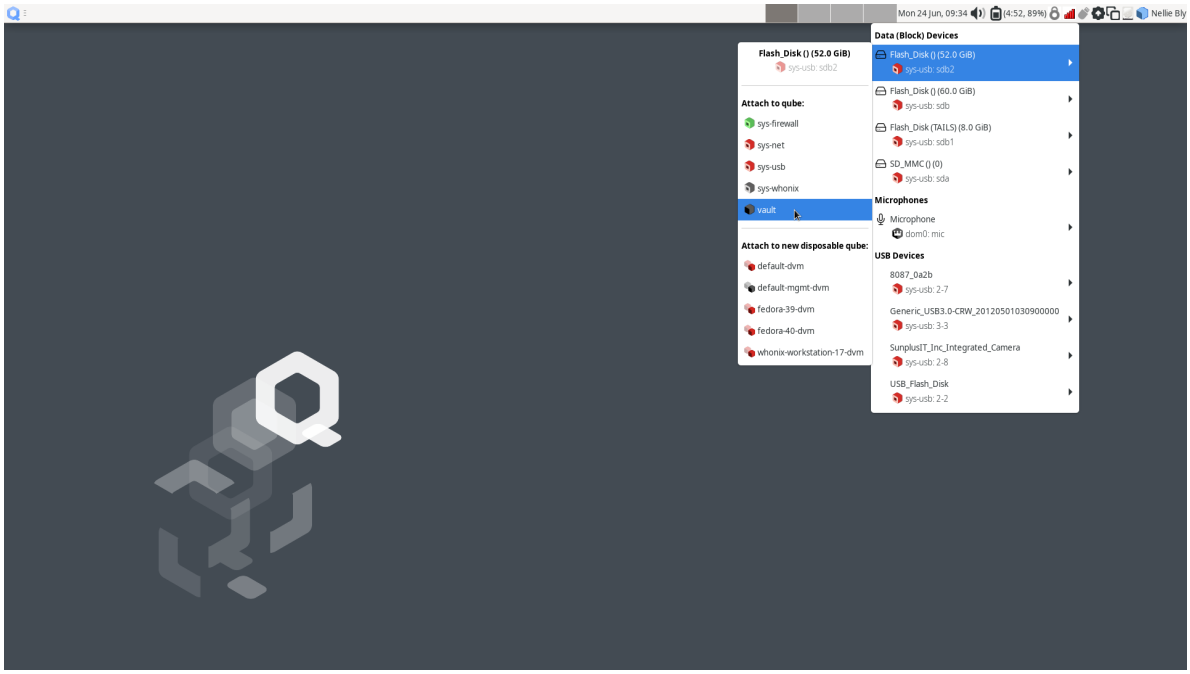
Import *Submission Private Key*

In order to decrypt submissions, you will need a copy of the *Submission Private Key* from your SecureDrop instance's *Secure Viewing Station*.

To protect this key and preserve the air gap, you will need to connect the *Secure Viewing Station* USB flash drive to a Qubes VM with no network access, and copy it from there to `dom0`. You cannot directly copy and paste to the `dom0` VM from another VM - instead, follow the steps below:

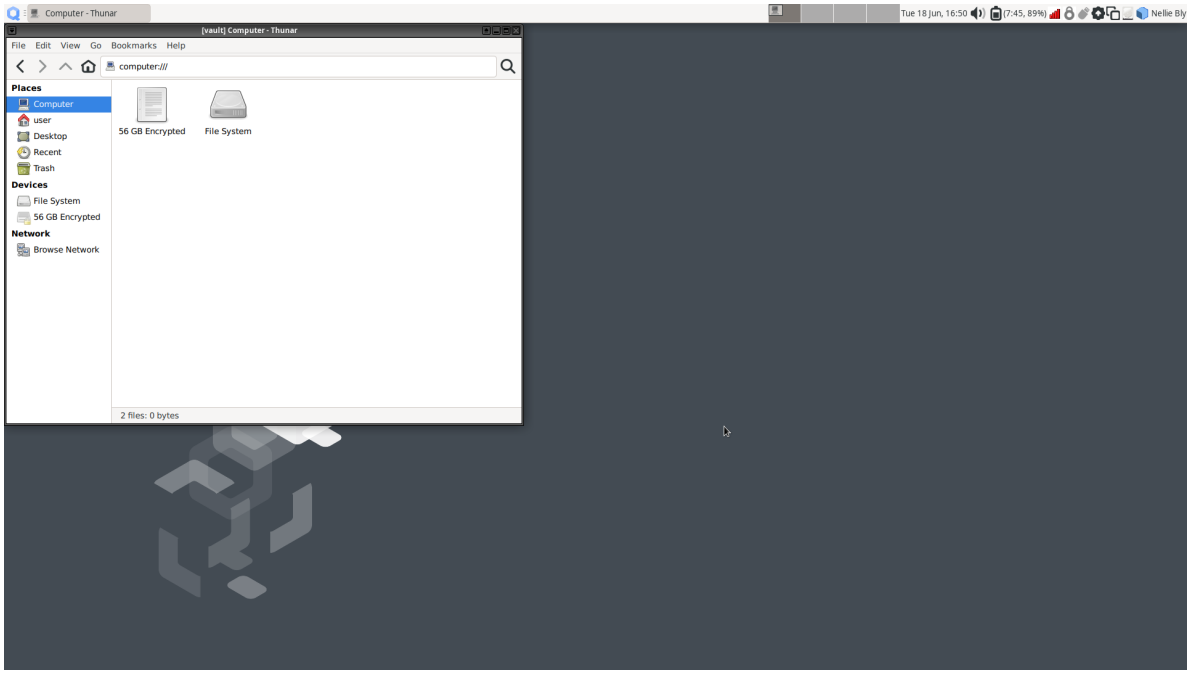
- First, use the network manager widget in the upper right panel to disable your network connection. These instructions refer to the `vault` VM, which has no network access by default, but if the *Secure Viewing Station* is attached to another VM by mistake, this will offer some protection against exfiltration.

- Next, choose ▶ **Apps** ▶ **vault** ▶ **Thunar File Manager** to open the file manager in the `vault` VM.
- Connect the *Secure Viewing Station* USB flash drive to a USB port on the Qubes computer, then use the devices widget in the upper right panel to attach it to the `vault` VM. There will be three entries for the USB flash drive in the section titled **Data (Block) Devices**. Choose the *unlabeled* entry (*not* the one labeled "TAILS") annotated with a `sys-usb` text that ends with a number, like `sys-usb:sdb2`. That is the persistent volume.



- In the the `vault` file manager, select the persistent volume's listing in the lower left sidebar. It will be named `N GB encrypted`, where N is the size of the persistent volume. Enter the *Secure Viewing Station* persistent volume passphrase to unlock and mount it. When asked if you would like to forget the password immediately or remember it until you logout, choose the option to **Forget password immediately**.

> **Note**
>
> You will receive a message that says **Failed to open directory "TailsData"**. This is normal behavior and will not cause any issues with the subsequent steps.



- Open a `dom0` terminal via ▶ ▶ **Other** ▶ **Xfce Terminal**. Once the terminal window opens, run the following command to import the *Submission Private Key*:

```
sdw-admin --configure
```

Follow the command prompts to complete *Submission Private Key* import.

> **Note**
>
> If there are multiple keys present on the device, `sdw-admin --configure` will print the fingerprints of those keys for you to select which to use as the *Submission Private Key*. You can open `<source interface address>.onion/metadata` in Tor Browser on another network-connected computer to check the correct key fingerprint used by your SecureDrop instance.

- Once the *Submission Private Key* import is complete, in the `vault` file manager, right-click on the **TailsData** sidebar entry, then select **Unmount** and disconnect the *Secure Viewing Station* USB flash drive.
- If you were prompted for a passphrase during import, you will now need to remove the passphrase on `sd-journalist.sec`. See *Removing the passphrase from a GPG key*.

Import *Journalist Interface* details

SecureDrop Workstation connects to your SecureDrop instance's API via the *Journalist Interface*. In order to do so, it will need the *Journalist Interface* address and authentication info. As the clipboard from another VM cannot be copied into `dom0` directly, follow these steps to copy the file into place:

- Locate a Tails-based *Admin Workstation* or *Journalist Workstation* USB flash drive. Both hold the address and authentication info for the *Journalist Interface*; if you also want to copy the *Journalist*'s password database, use the *Journalist Workstation* USB flash drive.
- Connect the USB flash drive to a USB port on the Qubes computer, then use the devices widget in the upper right panel to attach it to the `vault` VM. There will be 3 listings for the USB flash drive in the widget: one for

the base drive, one for the Tails partition (labeled `Tails`), and a 3rd unlabeled listing (for the persistent volume). Choose the third listing.

- In the the `vault` file manager, select the persistent volume's listing in the lower left sidebar. It will be named `N GB encrypted`, where N is the size of the persistent volume. Enter the persistent volume passphrase to unlock and mount it. When prompted, select the option to **Forget password immediately**.
- In the `dom0` terminal, proceed with the next import step of the `sdw-admin` command or re-run

```
sdw-admin --configure
```

The command will print out the imported *Journalist Interface* details to confirm before proceeding.

- If you used a Tails-based *Admin Workstation* drive, or you don't intend to copy a password database to this workstation, safely disconnect the USB flash drive now. In the `vault` file manager, right-click on the **TailsData** sidebar entry, then select **Unmount** and disconnect the USB flash drive.

1.35.13 Copy SecureDrop login credentials

When launching SecureDrop Inbox must enter their username, passphrase and two-factor code to connect with the SecureDrop server. You can manage these passphrases using the KeePassXC password manager in the `vault` VM. If this laptop will be used by more than one *Journalist*, we recommend that you shut down the `vault` VM now (using the Qube widget in the upper right panel), skip this section, and use a smartphone password manager instead.

In order to set up KeePassXC for easy use:

- Add KeePassXC to the application menu by selecting it from the list of available apps in  ▶ **Apps** ▶ **vault** ▶ **Settings** ▶ **Applications** and pressing the button labeled `>` (do not press the button labeled `>>`, which will add *all* applications to the menu).
- Launch KeePassXC via  ▶ **Apps** ▶ **vault** ▶ **KeePassXC**. When prompted to enable automatic updates, decline. `vault` is networkless, so the built-in update check will fail; the app will be updated through system updates instead.
- Close the application.

Important

The password database from the Tails-based *Admin Workstation* contains sensitive credentials not required by *Journalists*. Make sure to copy the credentials from the Tails-based *Journalist Workstation* USB flash drive.

In order to copy a *Journalist*'s login credentials:

- If a Tails-based *Journalist Workstation* USB flash drive is not currently attached, connect it, attach it to the `vault` VM, open it in the file manager, and enter its encryption passphrase.
- Locate the password database. It should be in the `Persistent` directory, and will typically be named `keepassx.kdbx` or similar.
- Open a second `vault` file manager window (`Ctrl + N` in the current window) and navigate to the **Home** directory.
- Drag and drop the password database to copy it.
- In the `vault` file manager, right-click on the **TailsData** sidebar entry, then select **Unmount** and disconnect the *Journalist Workstation* USB. Close this file manager window.

- In the file manager window that displays the home directory, open the copy you made of the password database by double-clicking it.
- If the database is passwordless, KeePassXC may display a security warning when opening it. To preserve convenient passwordless access, you can protect the database using a key file, via **Database ▶ Database settings ▶ Security ▶ Add additional protection ▶ Add Key File ▶ Generate**. This key file has to be selected when you open the database, but KeePassXC will remember the last selection.
- Inspect each section of the password database to ensure that it contains only the information required by the *Journalist* to log in.
- Close the application window and shut down the `vault` VM (using the Qube widget in the upper right panel). At this time, you can also re-enable the network connection using the network manager widget.

Manually importing from Tails drives

1.35.14 Manually import *Submission Private Key*

If importing the *Submission Private Key* using `sdw-admin --configure` fails, you can also copy the *Submission Private Key* manually.

- Open a `dom0` terminal via  ▶  ▶ **Other Tools ▶ Xfce Terminal**. Once the terminal window opens, run the following command to list the *Submission Private Key* details, including its fingerprint:

```
qvm-run --pass-io vault \
  "gpg --homedir /run/media/user/TailsData/gnupg -K --fingerprint"
```

- Next, run the comand:

```
qvm-run --pass-io vault \
  "gpg --homedir /run/media/user/TailsData/gnupg --export-secret-keys --armor
↪<SVSFingerprint>" \
  > /tmp/sd-journalist.sec
```

  where `<SVSFingerprint>` is the *Submission Private Key* fingerprint, typed as a single unit without whitespace. This will copy the *Submission Private Key* in ASCII format to a temporary file in dom0, `/tmp/sd-journalist.sec`.

- Verify the that the file starts with `-----BEGIN PGP PRIVATE KEY BLOCK-----` using the command:

```
head -n 1 /tmp/sd-journalist.sec
```

- Unmount the *Secure Viewing Station* USB flash drive.
- Run the following command in the `dom0` terminal:

```
sudo cp /tmp/sd-journalist.sec /usr/share/securedrop-workstation-dom0-config/
```

- You can run `sdw-admin --configure` to now import the *Journalist Interface* details and complete configuration.

  Alternatively, follow the steps below to do so manually. Once both *Submission Private Key* and *Journalist Interface* details are imported, proceed with *configuring the workstation*.

1.35.15 Manually import *Journalist Interface* details

If importing the *Journalist Interface* details using `sdw-admin --configure` fails, you can copy the configuration file to `dom0` manually.

- If your *Journalist Interface* is based on SecureDrop 2.13.0 or later, use the following command:

```
qvm-run --pass-io vault \
  "cat /run/media/user/TailsData/securedrop-admin/app-journalist.auth_private" \
  > /tmp/journalist.txt
```

- If your *Journalist Interface* is based on SecureDrop 2.12.10 or earlier, use the following command:

```
qvm-run --pass-io vault \
        "cat /run/media/user/TailsData/Persistent/securedrop/install_files/ansible-
↪base/app-journalist.auth_private" \
  > /tmp/journalist.txt
```

- Verify that the `/tmp/journalist.txt` file on `dom0` contains valid configuration information using the command `cat /tmp/journalist.txt` in the `dom0` terminal.
- Proceed with *configuring the workstation*

If you encounter a validation error due to a password-protected GPG key, see *Removing the passphrase from a GPG key*.

Once the *Journalist Interface* details and *Submission Private Key* have been copied to `dom0`, you can create the configuration for the SecureDrop Workstation.

- Your *Submission Private Key* has a unique fingerprint required for the configuration. Obtain the fingerprint by using this command:

```
gpg --with-colons --import-options import-show --dry-run --import /tmp/sd-
↪journalist.sec
```

  The fingerprint will be on a line that starts with `fpr`. For example, if the output included the line `fpr:::::::::65A1B5FF195B56353CC63DFFCC40EF1228271441:`, the fingerprint would be the character sequence `65A1B5FF195B56353CC63DFFCC40EF1228271441`.

- Next, create the SecureDrop Workstation configuration file:

```
cd /usr/share/securedrop-workstation-dom0-config
sudo cp config.json.example config.json
```

- The `config.json` file must be updated with the correct values for your instance. Open it with root privileges in a text editor such as `vi` or `nano` and update the following fields' values:
  - **submission_key_fpr**: use the value of the *Submission Private Key* fingerprint as displayed above
  - **hidserv.hostname**: use the hostname of the *Journalist Interface*, including the `.onion` TLD
  - **hidserv.key**: use the private v3 *Onion Service* authorization key value
  - **environment**: use the value `prod`

Note

You can find the values for the **hidserv.*** fields in the `/tmp/journalist.txt` file that you created in `dom0` earlier. The file will be formatted as follows:

```
ONIONADDRESS:descriptor:x25519:AUTHTOKEN
```

- Verify that the configuration is valid using the command below in the `dom0` terminal:

```
sdw-admin --validate
```

1.35.16 Install and configure SecureDrop Inbox

- These steps should be performed from a `dom0` terminal. **Start a dom0 terminal** via ▶ ▶ **Other Tools ▶ Xfce Terminal**.
- Configure infinite scrollback for your terminal via **Edit ▶ Preferences ▶ General ▶ Unlimited scrollback**. This helps to ensure that you will be able to review any error output printed to the terminal during the installation.
- Finally, in the `dom0` terminal, run the command:

```
sdw-admin --apply
```

This command will take a considerable amount of time and approximately 4GB of bandwidth, as it sets up multiple VMs and installs supporting packages. When the command finishes, reboot the machine to complete the installation. This SecureDrop Workstation is finally ready to use!

1.35.17 Test the *Admin Workstation*

The preflight updater will start automatically after logging into the system. Please follow the preflight updater's instructions.

> **Note**
>
> If you close SecureDrop Inbox during your session, you can launch it again using the SecureDrop icon on the desktop.

Once the update check is complete, the SecureDrop Client will launch. Log in using an existing journalist account and verify that *Sources* are listed and submissions can be downloaded, decrypted, and viewed.

1.35.18 Enable password copy and paste

If you use KeePassXC in the `vault` VM to manage login credentials, you can enable the user to copy passwords to SecureDrop Inbox using inter-VM copy and paste. While this is relatively safe, we recommend reviewing the section *Managing Clipboard Access* of this guide, which goes into further detail on the security considerations for inter-VM copy and paste.

The password manager runs in the networkless `vault` VM, and the SecureDrop Inbox application runs in the `sd-app` VM. To permit this one-directional clipboard use, issue the following command in `dom0`:

```
qvm-tags vault add sd-send-app-clipboard
```

Confirm that the tag was correctly applied using the `ls` subcommand:

```
qvm-tags vault ls
```

To revoke this configuration change later or correct a typo, you can use the `del` subcommand, e.g.:

```
qvm-tags vault del sd-send-app-clipboard
```

Troubleshooting `sdw-admin`

1.35.19 "Failed to return clean data"

An error similar to the following may be displayed during an installation or update:

```
sd-log:
      ----------
      _error:
          Failed to return clean data
      retcode:
          None
      stderr:
      stdout:
          deploy
```

This is a transient error that may affect any of the SecureDrop Workstation VMs. To clear it, run the installation command or update again.

1.35.20 "Temporary failure resolving"

Transient network issues may cause an installation to fail. To work around this, verify that you have a working Internet connection, and re-run the `sdw-admin --apply` command.

1.36 Migrating a *Journalist Workstation*

1.37 Removing the passphrase from a GPG key

GPG key files should not be passphrase-protected for use with SecureDrop Workstation.

In a `dom0` terminal on your Qubes workstation ( ►  ► **Other** ► **Xfce Terminal**), assuming a passphrase-protected secret key file `/tmp/sd-journalist.sec`, import the key into a new temporary GnuPG directory, entering the passphrase when prompted:

```
export GPGTMP=`mktemp -d`     # create a tempdir
gpg --homedir=${GPGTMP} --pinentry=loopback --import /tmp/sd-journalist.sec
```

Next, check the key id:

```
gpg --homedir=${GPGTMP} --list-secret-keys --keyid-format=long
```

The output should list the key with a line similar to:

```
sec   rsa4096/XXXXXXXXXX <creation date>
```

The `XXXXXXXXXX` value is the key id, which you can use to open the key in edit mode with the following command:

```
gpg --homedir=${GPGTMP} --pinentry=loopback --edit-key XXXXXXXXXX
```

In the GPG interactive prompt, enter the command `passwd` to change the passphrase. You will first be prompted for the current passphrase. Then, on the next prompt, press Enter for a new blank passphrase, and Enter again when prompted to repeat it. Then exit with the command `quit`.

You should now have a passphrase-less version of the key in the $GPGTMP keyring. To export it, use the following command with the same key id as above:

```
gpg --homedir=${GPGTMP} --export-secret-key --armor XXXXXXXXXX > /tmp/nopassphrase.sec
```

Verify that the new keyfile `/tmp/nopassphrase.sec` starts with the `-----BEGIN PGP PRIVATE KEY BLOCK-----` line. Copy the key into place:

```
sudo cp /tmp/nopassphrase.sec /usr/share/securedrop-workstation-dom0-config/sd-
↪journalist.sec
```

If you are provisioning SecureDrop Workstation for the first time, continue with the installation instructions. Or, to re-check an existing configuration:

```
sdw-admin --validate
```

1.38 Onboard *Journalists*

At this point, the only person who has access to the system is the admin. In order to grant access to *Journalists*, you will need to do some additional setup for each individual *Journalist*.

1.38.1 Provision *Journalist Workstation*

1.38.2 Add an account on the *Journalist Interface*

Finally, you need to add an account on the *Journalist Interface* so the *Journalist* can log in and access submissions.

1.38.3 Adding users

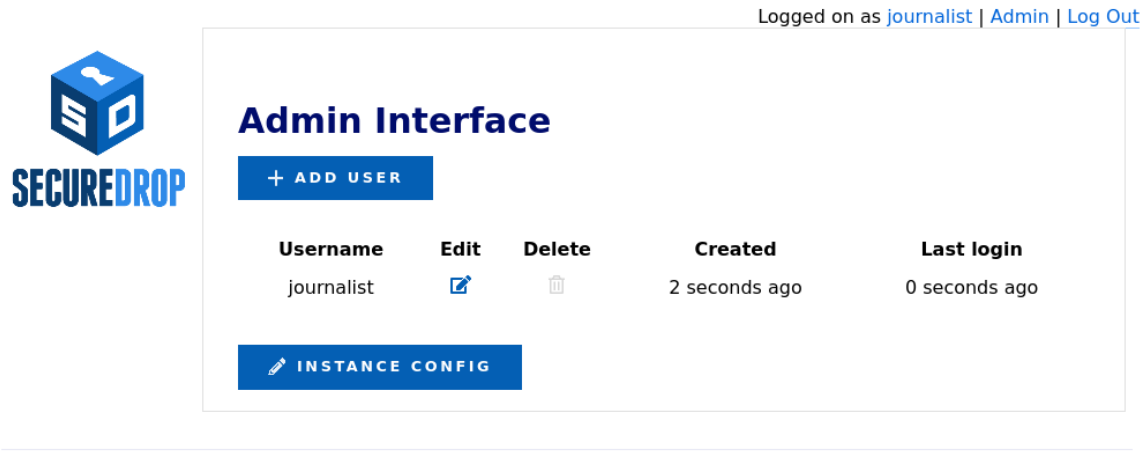
After logging in, you can add new user accounts for the *Journalists* at your organization who will be checking the system for submissions. Make sure the *Journalist* is physically in the same room as you when you do this, as they will have to be present to enable *Two-Factor Authentication*. SecureDrop supports the use of either a smartphone authenticator app or a Yubikey for *Two-Factor Authentication*. If an app is to be used, the *Journalist* should install it before proceeding with the account setup.

Tip

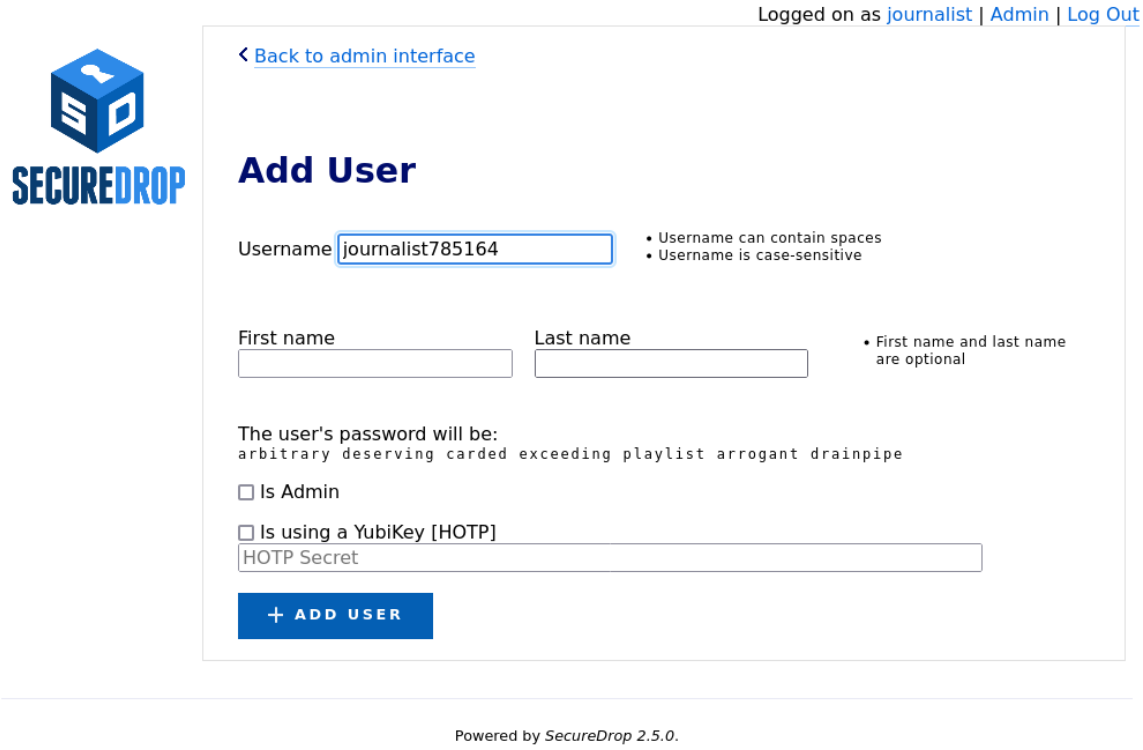
We recommend using FreeOTP (available for Android and for iOS) to generate two-factor codes because it is Free Software. However, if it does not work for you for any reason, alternatives exist:

- Google Authenticator for Android and iOS (proprietary)
- authenticator for the desktop (Free Software)

1. Click **Admin** in the top right corner of the page to load the *Admin Interface*.



2. Click **Add User** to add a new user.



3. Hand the keyboard over to the *Journalist* so they can create their own username.
4. Once they're done entering a username for themselves, have them save their pre-generated Diceware passphrase to their password manager.
5. If the new account should also have admin privileges, allowing them to add or delete other journalist accounts, select **Is Admin**.
6. Finally, set up *Two-Factor Authentication* for the account, following one of the two procedures below for your chosen method.

Note

The username **deleted** is reserved, as it is used to mark accounts which have been deleted from the system.

FreeOTP

1. If the *Journalist* is using FreeOTP or another app for *Two-Factor Authentication*, click **Add User** to proceed to the next page.

2. Next, the *Journalist* should open FreeOTP on their smartphone and scan the barcode displayed on the screen.
3. If they have difficulty scanning the barcode, they can tap on the icon at the top that shows a plus and the symbol of a key and use their phone's keyboard to input the two-factor secret into the `Secret` input field, without whitespace.
4. Inside the FreeOTP app, a new entry for this account will appear on the main screen, with a six-digit number that recycles to a new number every thirty seconds. The *Journalist* should enter the six-digit number in the **Verification code** field at the bottom of the **Enable FreeOTP** form and click **Submit**.

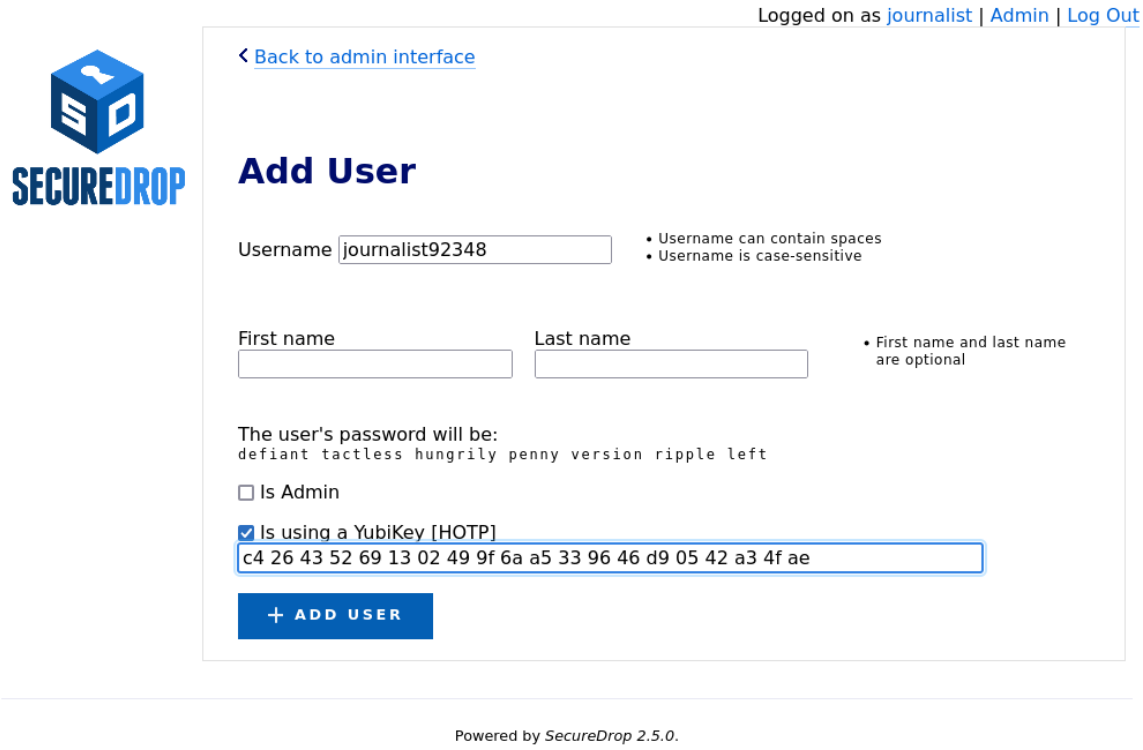
If *Two-Factor Authentication* was set up successfully, you will be redirected back to the *Admin Interface* and will see a confirmation that the two-factor code was verified.

Note

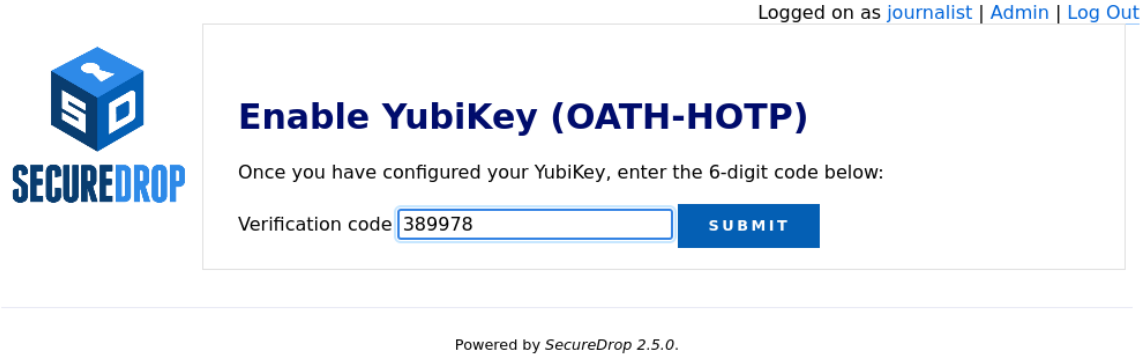
If the QR code for setting up *Two-Factor Authentication* in your mobile authenticator app is not displayed, it may be blocked by Tor Browser. You can set Tor Browser's security level to **Standard** by clicking on the Shield icon. Alternatively, you can manually type in the two-factor secret (in FreeOTP, use the **Add token** option from the menu).

YubiKey

1. If the *Journalist* wishes to use a YubiKey for *Two-Factor Authentication*, select **Is using a YubiKey**. You will then need to enter their YubiKey's OATH-HOTP Secret Key. For more information on how to retrieve this key, read the *YubiKey Setup Guide*.



2. Once you've entered the Yubikey's OATH-HOTP Secret Key, click **Add User**. On the next page, have the *Journalist* authenticate using their YubiKey, by inserting it into a USB port on the workstation and pressing its button.



3. If everything was set up correctly, you will be redirected back to the *Admin Interface*, where you should see a flashed message that says "The two-factor code for user *new username* was verified successfully.".

The *Journalist* will require their username, passphrase, and *Two-Factor Authentication* method whenever they check SecureDrop. Make sure that they have memorised their username and passphrase, or stored them in their password manager, and that they can keep their *Two-Factor Authentication* device secure.

1.38.4 Verify *Journalist* setup

1.39 Deployment overview

Once SecureDrop is installed on a news organization's servers, it's important for the administrator to configure it in a way that provides the greatest protection for *Sources* and *Journalists*, given the unique needs and constraints of the organization.

The deployment section here covers a variety of tasks an administrator might need to perform to successfully deploy SecureDrop, depending on organizational needs and requirements.

Certain topics, such as creating a *Landing Page* and onboarding *Journalists*, are universal to all SecureDrop instances. Other topics are optional, and are only needed if they fit in with the organization's security policies and newsroom workflows.

The deployment tasks generally only need to be performed once. For tasks related to the upkeep and troubleshooting of your SecureDrop instance, we recommend reviewing *the maintenance documentation.*

1.40 Protecting the security of the system

SecureDrop is only as secure as the environment that surrounds it. To keep *Sources* safe, the news organization's website, physical space, and dedicated SecureDrop hardware must employ a set of basic security best practices or risk losing any source protection provided by SecureDrop.

Freedom of the Press Foundation eventually plans to list all of those SecureDrop onion addresses that meet the minimum requirements for deployment best practices as "verified" on its website. If your organization cannot follow the minimum guidelines, we cannot recommend your SecureDrop instance as safe to use.

In addition to implementing the following best practices, we strongly recommend that you have a reputable security firm perform a review of your organization's public website prior to launching an instance of SecureDrop. Upon request, we can help put you in touch with a few security firms if you need more assistance.

1.41 *Landing Page*

SecureDrop itself runs as a Tor *Onion Service*. Organizations also need to create a SecureDrop *Landing Page* that will:

- explain how SecureDrop works
- give *Sources* instructions on how to access the Tor *Onion Service*
- disclose the risks of accessing the SecureDrop instance or submitting documents

We also recommend including a privacy policy (see our *Sample SecureDrop privacy policy*) describing what data is collected and how it will be used by your organization.

Note

SecureDrop will bring more attention to your organization from security researchers and others. A *Landing Page* that fails to implement minimum security requirements is sure to be noticed, and could undermine trust, discouraging possible *Sources*.

1.41.1 *Landing Page* content suggestions

The content below presents sample text for the SecureDrop component of a news organization's tips page. It does not account for any specific legal or organizational needs, but should provide guidance for any outlet getting started on crafting *Landing Page* language. Any tweaks to the sample content should be left to the legal and editorial discretion of the individual outlet, and should be viewed as essential to upholding source protection and transparency.

What is SecureDrop?

SecureDrop is an anonymity tool for journalists and whistleblowers. As a source, you can use our SecureDrop installation to anonymously submit documents to our organization. Our journalists use SecureDrop to receive source materials and securely communicate with anonymous contacts.

What should I know before submitting material through SecureDrop?

To protect your anonymity when using SecureDrop, it is essential that you do not use a network or device that can easily be traced back to your real identity. Instead, use public wifi networks and devices you control.

- Do NOT access SecureDrop on your employer's network.
- Do NOT access SecureDrop using your employer's hardware.
- Do NOT access SecureDrop on your home network.
- DO access SecureDrop on a network not associated with you, like the wifi at a library or cafe.

Got it. How can I submit files and messages through SecureDrop?

Once you are connected to a public network at a cafe or library, download and install the desktop version of Tor Browser.

Launch Tor Browser. Visit our organization's unique SecureDrop URL at **http://our-unique-URL.onion/**. Follow the instructions you find on our source page to send us materials and messages.

When you make your first submission, you will receive a unique codename. Memorize it. If you write it down, be sure to destroy the copy as soon as you've committed it to memory. Use your codename to sign back in to our source page, check for responses from our journalists, and upload additional materials.

As a source, what else should I know?

No tool can absolutely guarantee your security or anonymity. The best way to protect your privacy and anonymity as a source is to adhere to best practices.

You can use a separate computer you've designated specifically to handle the submission process. Or, you can use an alternate operating system like Tails, which boots from a USB flash drive and erases your activity at the end of every session.

A file contains valuable metadata about its source — when it was created and downloaded, what machine was involved, the machine's owner, etc. You can scrub metadata from some files prior to submission using the Metadata Anonymization Toolkit featured in Tails.

Your online behavior can be extremely revealing. Regularly monitoring our publication's social media or website can potentially flag you as a source. Take great care to think about what your online behavior might reveal, and consider using Tor Browser to mitigate such monitoring.

Our organization retains strict access control over our SecureDrop project. A select few journalists within our organization will have access to SecureDrop submissions. We control the servers that store your submissions, so no third party has direct access to the metadata or content of what you send us.

Do not discuss leaking or whistleblowing, even with trusted contacts.

1.41.2 The SecureDrop directory

SecureDrop maintains a directory of instances that meet our strict guidelines. If you would like to be considered for inclusion in this directory, make sure your *Landing Page* features the necessary items from the sample above, and is in compliance with the technical requirements below, then send us a request using this form.

There are several benefits to being included in the SecureDrop directory. The most significant benefit is that it will be easier for potential *Sources* to find your SecureDrop instance. Additionally, being included in the directory makes you eligible for *an onion name.* This improves the experience by turning a lengthy, non-descriptive address into one that is short and memorable. For example, a long onion address might look like:

```
sdolvtfhatvsysc6l34d65ymdwxcujausv7k5jk4cy5ttzhjoi6fzvyd.onion
```

whereas the shorter onion name might look like:

```
nyworld.securedrop.tor.onion
```

If you wish to receive an onion name, one can be provided during the instance verification process. The format for short onion addresses is:

```
organization.securedrop.tor.onion
```

where `organization` can be any name you request, within reason.

Being included in the SecureDrop directory may make your instance more visible, which could result in an uptick of illegitimate (spam) submissions. If you notice an increase in spam after being included in the directory, please let us know and we can remove your instance from the directory.

1.41.3 URL and location

Your *Landing Page* must be a path at your top-level domain, e.g. organization.com/securedrop, rather than a subdomain (e.g., securedrop.organization.com). This is because DNS and TLS do not always encrypt the hostname, so a SecureDrop user whose connection is being monitored would be trivially discovered if you were to use a subdomain.

If the *Landing Page* is deployed on the same domain as another site, you might consider having some specific configuration (such as the security headers below) apply only to the /securedrop request URI. This can be done in Apache by the encapsulating these settings within a <Location> block, which can be defined similarly in nginx by using the location { } directive.

Warning

Except for rare extenuating circumstances, this is a requirement for inclusion in the SecureDrop Directory

1.41.4 HTTPS only (no mixed content)

HTTPS encryption is the number-one security requirement for your site's SecureDrop *Landing Page*. Without HTTPS, a *Source* can easily be exposed as a visitor to your site.

This may be difficult if your website serves advertisements or utilizes a legacy content delivery network. You should make sure the SecureDrop *Landing Page* does not serve ads of any kind, even if the rest of your site does.

If you do not serve ads on any of your site, you should also consider switching your whole site over to HTTPS by default immediately. If you do serve ads, consider pressuring your ad networks to enable you to switch to HTTPS for your entire website in the future.

If your website needs to operate in both HTTPS and HTTP mode, use protocol-relative URLs for resources such as images, CSS and JavaScript in common templates to ensure your page does not end up in a mixed HTTPS/HTTP state.

Consider submitting your domain to be included in the Chrome HSTS preload list if you can meet all of the requirements. This will tell web browsers that the site is only ever to be reached over HTTPS.

Warning

This is a strict requirement for inclusion in the SecureDrop Directory

1.41.5 Perfect forward secrecy

Perfect forward secrecy (PFS) is a property of encryption protocols that ensures each SSL session has a unique key, meaning that if the key is compromised in the future it can't be used to decrypt previously recorded SSL sessions. You may need to talk to your CA (certificate authority) and CDN (content delivery network) for this, although our recommended configuration below provides forward secrecy.

1.41.6 SSL certificate recommendations

Regardless of where you choose to purchase your SSL cert and which CA issues it, you'll often be asked to generate the private key and a CSR (certificate signing request).

When you do this, it's imperative that you use SHA-2 as the hashing algorithm instead of SHA-1, which is being phased out. You should also choose a key size of *at least* 2048 bits. These parameters will help ensure that the encryption used on your *Landing Page* is sufficiently strong. The following example OpenSSL command will create a private key and CSR with a 4096-bit key length and a SHA-256 signature:

```
openssl req -new -newkey rsa:4096 -nodes -sha256 -keyout domain.com.key -out domain.com.
↪csr
```

Don't load any resources (scripts, web fonts, etc.) from third parties (e.g. Google Web Fonts)

This will potentially leak information about *Sources* to third parties, which can more easily be accessed by law enforcement agencies. Simply copy them to your server and serve them yourself to avoid this problem.

1.41.7 Do not use third-party analytics, tracking, or advertising

Most news websites, even those that are non-profits, use third-party analytics tools or tracking bugs on their websites. It is vital that these are disabled for the SecureDrop *Landing Page*.

In the past, some news organizations were heavily criticized when launching their SecureDrop instances because their *Landing Page* contained trackers. They claimed they were going to great lengths to protect *Sources*' anonymity, but by having trackers on their *Landing Page*, this also opened up multiple avenues for third parties to collect information on those *Sources*. This information can potentially be accessed by law enforcement or intelligence agencies and could unduly expose a *Source*.

Similarly, consider avoiding Cloudflare (and other CDNs like Akamai, StackPath, Incapsula, Amazon CloudFront, etc.) for the SecureDrop *Landing Page*. These services intercept requests between a potential *Source* and the SecureDrop *Landing Page* and can be used to track or collect information on *Sources*.

Warning

This is a strict requirement for inclusion in the SecureDrop Directory

1.41.8 Do not hyperlink onion addresses

Because a visitor to your *Landing Page* may not be using Tor Browser yet, clicking a link to your SecureDrop instance or to any other onion address may result in an error message. Worse, depending on the browser and network configuration, it may cause lookups that an adversary can use to identify SecureDrop-related behavior.

Instead, we recommend including onion addresses in plain text, without a hyperlink.

If you have been provided a short onion name for your instance, this address will also need to be plain text, without a hyperlink. We recommend using the text below to provide maximum clarity:

```
The SecureDrop instance can be found by entering the following address in
the desktop version of Tor Browser: <short onion name>

Alternately, you can access the instance by entering: <long onion address>
```

Warning

This is a strict requirement for inclusion in the SecureDrop Directory

1.41.9 Avoid direct links to securedrop.org

We appreciate that you may want to link to the SecureDrop website to give *Landing Page* visitors more information about the system. Unfortunately, if a visitor visits these links without using Tor Browser, this generates traffic that an adversary may be able to use to identify SecureDrop-related behavior, regardless of the use of HTTPS.

We suggest offering a reference to the SecureDrop *Onion Service* in plain text, without a hyperlink (as per the preceding section):

sdolvtfhatvsysc6l34d65ymdwxcujausv7k5jk4cy5ttzhjoi6fzvyd.onion

Warning

This is a strict requirement for inclusion in the SecureDrop Directory

1.41.10 Apply security headers

Security headers give instructions to the web browser on how to handle requests from the web application. These headers set strict rules for the browser and help mitigate against potential attacks. Given the browser is a main avenue for attack, it is important these headers are as strict as possible.

You can use the site securityheaders.com to easily test your website's security headers.

If you use Apache, you can use these:

```
Header set Cache-Control "max-age=0, no-cache, no-store, must-revalidate"
Header edit Set-Cookie ^(.*)$ $;HttpOnly
Header set Pragma "no-cache"
Header set Expires "-1"
```

(continues on next page)

(continued from previous page)

```
Header always append X-Frame-Options: DENY
Header set X-XSS-Protection: "1; mode=block"
Header set X-Content-Type-Options: nosniff
Header set X-Download-Options: noopen
Header set X-Permitted-Cross-Domain-Policies: master-only
Header set Content-Security-Policy: "default-src 'none'; script-src 'self'; style-src
↪'self'; img-src 'self'; font-src 'self';"
Header set Referrer-Policy "no-referrer"
Header set Permissions-Policy "camera 'none'; display-capture 'none'; geolocation 'none';
↪ microphone 'none'; payment 'none'; usb 'none';"
```

If you intend to run nginx as your webserver instead, this will work:

```
add_header Cache-Control "max-age=0, no-cache, no-store, must-revalidate";
add_header Pragma no-cache;
add_header Expires -1;
add_header X-Frame-Options DENY;
add_header X-XSS-Protection "1; mode=block";
add_header X-Content-Type-Options nosniff;
add_header X-Download-Options noopen;
add_header X-Permitted-Cross-Domain-Policies master-only;
add_header Content-Security-Policy "default-src 'none'; script-src 'self'; style-src
↪'self'; img-src 'self'; font-src 'self';";
add_header Referrer-Policy "no-referrer";
add_header Permissions-Policy "camera 'none'; display-capture 'none'; geolocation 'none';
↪ microphone 'none'; payment 'none'; usb 'none';";
```

1.41.11 Additional apache configuration

To enforce HTTPS/SSL always, you need to set up redirection within the HTTP (port 80) virtual host:

```
RewriteEngine On
RewriteCond %{HTTPS} off
RewriteRule (.*) https://%{HTTP_HOST}%{REQUEST_URI}
```

The same thing can be achieved in nginx with a single line:

```
return 301 https://$server_name$request_uri;
```

In your SSL (port 443) virtual host, set up HSTS and use these settings to give preference to the most secure cipher suites:

```
Header set Strict-Transport-Security "max-age=16070400;"
SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1
SSLHonorCipherOrder on
SSLCompression off
SSLCipherSuite EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH
```

Here's a similar example for nginx:

```
add_header Strict-Transport-Security max-age=16070400;
ssl_protocols TLSv1.2;
```

(continues on next page)

(continued from previous page)

```
ssl_prefer_server_ciphers on;
ssl_ciphers "EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH";
```

Here's a similar example for nginx if the system supports TLS 1.3:

```
add_header Strict-Transport-Security max-age=16070400;
ssl_protocols TLSv1.2 TLSv1.3;
ssl_prefer_server_ciphers on;
ssl_ciphers "TLS-CHACHA20-POLY1305-SHA256:TLS-AES-256-GCM-SHA384:TLS-AES-128-GCM-
↪SHA256:EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH";
```

Note

We have prioritized security in selecting these cipher suites, so if you choose to use them then your site might not be compatible with legacy or outdated browsers and operating systems. For a good reference check out Mozilla's recommendations.

You'll need to run `a2enmod headers ssl rewrite` for all these to work. You should also set `ServerSignature Off` and `ServerTokens Prod`, typically in /etc/apache2/conf.d/security. For nginx, use `server_tokens off;` so that the webserver doesn't leak extra information.

If you use nginx, you can follow this link and use the configuration example provided by ProPublica.

Warning

Setting the `Referrer-policy` header to `no-referrer` is a strict requirement for inclusion in the SecureDrop directory. Setting the remaining headers as described is strongly recommended, but will be reviewed on a case-by-case basis for inclusion in the directory and does not necessarily prevent the instance from being included.

Set up change detection monitoring for the web application configuration and *Landing Page* content

If possible, you should set up monitoring to detect changes to the *Landing Page* and the configuration files of the web server hosting the page. If you do not have an existing monitoring system for your site, OSSEC is a free and open source host-based intrusion detection suite that includes a file integrity monitor. More information can be found here.

Note

We do not recommmend using the *Monitor Server* to monitor your *Landing Page*. It should be used for the *Application Server* only.

Don't log access to the *Landing Page* in the webserver

Here's an Apache example that would exclude the *Landing Page* from logging. However you still need to make sure no other assets get logged!

```
SetEnvIf Request_URI "^/securedrop($|(\/.*))" dontlog
CustomLog logs/access_log common env=!dontlog
```

In nginx, logging can be disabled by adding the following directives within the *Landing Page* `location {}` block:

```
access_log off;
error_log /dev/null;
```

1.41.12 Further security considerations

To guard your *Landing Page* against being modified by an attacker and directing *Sources* to a rogue SecureDrop instance, you will need good security practices applying to the machine where it is hosted. Whether it's a VPS in the cloud or dedicated server in your office, you should consider the following:

- Brute force login protection (see fail2ban or sshguard)
- Disable root SSH login
- Use SSH keys instead of passwords
- Use long, random and complex passwords
- Firewall rules to restrict accessible ports (see iptables or ufw)
- AppArmor, grsecurity, SELINUX, modsecurity
- Intrusion and/or integrity monitoring (see Logwatch, OSSEC, Snort, rkhunter, chkrootkit)
- Downtime alerts (Nagios or Pingdom)
- *Two-Factor Authentication* (see libpam-google-authenticator, libpam-yubico)

It's preferable for the *Landing Page* to have its own segmented environment instead of hosting it alongside other sites running potentially vulnerable software or content management systems. Check that user and group file permissions are locked down and that modules or gateway interfaces for dynamic scripting languages are not enabled. You don't want any unnecessary code or services running as this increases the attack surface.

1.41.13 How to test your *Landing Page* using Tor Browser

Sources may visit your *Landing Page* using Tor.

Many websites are configured with security measures that only apply when Tor is in use. For example, Tor visitors may be requested to solve a CAPTCHA, may trigger warnings that are specific to some Tor exit nodes, or may be unable to load the page altogether because of Tor-specific DDoS protections.

The effect of such measures cannot be tested without using Tor, and it is a very bad experience for a *Source* if visiting a *Landing Page* doesn't work as expected. Because of that, we **recommended strongly** that you test your organization's *Landing Page* using Tor *before* you start advertising it.

You can do so using Tor Browser:

1. Download Tor Browser from the Tor Project website.
2. Ensure the Tor Browser security level is set to "Safest" by clicking on the shield icon. If not, click "Settings…", then "Change…", then select "Safest". Finally, click "Save and restart" to re-launch the browser and apply the new settings.
3. Visit your *Landing Page*.
4. Verify that everything works as expected.
5. Reload the page using a different Tor circuit by clicking on "New Tor Circuit for this Site" in the site information menu (padlock icon in the URL bar) or in the hamburger menu.
6. Verify that everything still works as expected.
7. Repeat the previous two steps several times to test with exit nodes in different countries and regions.

1.42 Getting an onion name for your SecureDrop

1.42.1 What are onion names?

Onion names are short, memorable addresses that visitors can use to access an *Onion Service* (e.g., a news organization's SecureDrop) using Tor Browser.

Imagine a SecureDrop instance for a new organization called *The New York World* with an onion address like this:

sdolvtfhatvsysc6l34d65ymdwxcujausv7k5jk4cy5ttzhjoi6fzvyd.onion

An onion name for this SecureDrop instance could be:

nyworld.securedrop.tor.onion

The general format for a SecureDrop onion name is:

`<organization>.securedrop.tor.onion`

1.42.2 How they work

Onion names are supported in the desktop version of Tor Browser (introduced in version 9.5). The mapping between onion names and the full-length onion addresses is done using a custom, signed ruleset for SecureDrop instances maintained by Freedom of the Press Foundation. The ruleset is updated automatically by Tor Browser, and no information is sent to third party servers when contacting a SecureDrop using an onion name.

Onion names are currently not supported by the mobile version of Tor Browser, or by any other browser. (SecureDrop strongly recommends the use of the desktop version of Tor browser.)

The Tor project has committed to continued support of the onion name feature in some form. The underlying implementation and the address format may change in future iterations of this feature. To the extent that any changes are required, we will reach out to coordinate them with you.

1.42.3 Getting an onion name

Freedom of the Press Foundation maintains onion names for SecureDrop instances which:

- are using v3 *Onion Services*
- are part of the SecureDrop Directory

We will generally approve onion names that meaningfully correspond to your name or that of your organization. Please note that, to disambiguate organizations in different countries with the same name, we may request the addition of a country code (e.g. `<organization>.<country code>.securedrop.tor.onion`).

If your SecureDrop instance is not part of the directory yet, you can *begin the process here*. In order to be eligible for inclusion, your SecureDrop and its associated clearnet *Landing Page* must be set up consistent with the best practices recommended in our documentation.

If you are already part of the SecureDrop directory and would like a short onion name, *please contact us.*

1.42.4 Does This Replace the original address?

No, the onion name is only a human-friendly name for the full-length address. The original v3 address can continue to be used like normal, this just gives *Sources* an easier to remember option for accessing your SecureDrop.

We recommend that you list both the onion name and the full v3 address on your *Landing Page*. This allows *Sources* to verify both addresses against the information included in our directory, and also provides a fallback should the onion name fail to load for any reason.

Please note that the desktop version of Tor Browser is needed to access onion names, which is also generally our security recommendation.

1.42.5 Updating an onion name

If you wish to change or retire your onion name, please reach out to the SecureDrop team. In the event that you wish to completely retire your SecureDrop instance, we recommend that you contact us ahead of time if possible, so we can schedule the onion name update on the same day.

In any event, we will attempt to respond to any update request within 2 business days.

1.42.6 Revoking onion names

Onion names are tied to inclusion in the SecureDrop Directory. We may remove SecureDrop instances from the directory at our discretion for reasons including but not limited to:

- an instance is stuck on an old software version, and can no longer be considered secure;
- an instance is unreachable for extended periods of time;
- the configuration of an instance or the associated *Landing Page* differs substantially from our security recommendations in a manner that may put *Sources* at risk.

Unless the removal is an emergency, we will attempt to offer a substantial grace period prior to the revocation of an onion name, to ensure you can inform your *Sources* about the change to your onion address.

1.43 Whole site changes

Ideally, some or all of the following changes are made to improve the overall security of the path to the *Landing Page* and obfuscate traffic analysis.

1. Make your entire site available through HTTPS.
    - That way, visits to your *Landing Page* won't stand out as the only encrypted traffic to your site.
2. Include an iframe for all (or a random subset of) visitors, loading this particular URL (hidden).
    - By artificially generating traffic to the endpoint it will be harder to distinguish these from other, 'real' requests.
    - Use a random delay for adding the iframe (otherwise the 'pairing' with the initial HTTP request may distinguish this traffic).
3. Print the link, URL and info block on the dead trees (the paper), as others have suggested.
4. Add HSTS headers.

1.43.1 Suggested

- For publicly advertised SecureDrop instances display the *Source Interface*'s *Onion Service* onion address on all of the organization public pages.
- Mirror Tor Browser and Tails so *Sources* do not have to visit torproject.org to download it.

1.44 Sample SecureDrop privacy policy

[DATE]

SecureDrop strives to create a more secure environment for whistleblowers to give information to journalists. It was installed at **[MEDIA ORG]** with the help of Freedom of the Press Foundation.

Please read this privacy policy carefully. It explains what information what type of information SecureDrop does and does not collect, and why.

1.44.1 Collection of information from sources

- We don't ask or require you to provide any personally identifying information when you submit materials through SecureDrop.
- The system does not record your IP address, information about your browser, computer, or operating system. Furthermore, the SecureDrop pages do not embed third-party content or deliver persistent cookies to your browser.
- The server will only store the date and time of the newest message sent from each source. Once you send a new message, the time and date of your previous message is automatically deleted.
- Journalists decrypt and read each message offline. They are encouraged to delete messages from the server on a regular basis.
- Please keep in mind that the actual messages you send and receive through SecureDrop may include personally identifying information. For this reason, once you read a journalist's message, we recommend you delete it.

Also please note that when you submit certain types of files through SecureDrop, you may be sending us metadata associated with that file.

For example, if you submit a photo through SecureDrop in JPEG format, the file may include information about the date, time, and the GPS location of where it was taken, and the type of device used to take the photo. Similarly, if you submit a Word file (.doc or .docx) through SecureDrop, it may include the identity of the document's author, the author's operating system, GPS data about the author's location, and the date and time when the document was created.

Our policy is to scrub metadata from the files we receive through SecureDrop before publication. If you don't want to send us metadata, please use the Metadata Anonymization Toolkit to scrub the file before you submit it.

1.44.2 Collection of information about journalists' use of SecureDrop

[MEDIA ORG] collects information about journalists' use of SecureDrop for security monitoring and to make sure the system works properly.

This information we collect about journalists includes details about the device, browser, and operating system journalists use when accessing the system, and the date and time of each session.

We retain these access logs for **[___]** days, and then delete them.

1.44.3 Data security

[MEDIA ORG] works diligently to protect the identities of our sources and keep the information they give us confidential.

SecureDrop servers are under the physical control of **[MEDIA ORG]** and do not share common elements of the **[MEDIA ORG'S]** other infrastructure.

However, no one can truly guarantee 100% security of any system. Like all software, SecureDrop may contain bugs. Ultimately, you use the SecureDrop service at your own risk.

1.44.4 Children under 13

The Children's Online Privacy Protection Act restricts our ability to collect personal information from children under 13. This site is not directed to children 12 or younger.

1.44.5 Changes to this policy

We may revise this Privacy Policy from time to time. The most current version of the policy will govern our collection and use of personal information and will always be at **[LINK]**. If we make changes that we believe are material, we will prominently display a notice on our site **[___]** days before we make those changes.

1.44.6 Contact

[MEDIA ORG] welcomes questions, concerns, and feedback about this policy. If you have suggestions for us, feel free to let us know at **[EMAIL ADDRESS]**.

1.45 Promoting your SecureDrop instance

At Freedom of the Press Foundation, we've found news organizations that get the most out of SecureDrop are those who promote it regularly and effectively. SecureDrop will only be used by *Sources* if they know it exists, so it's best to promote its use in a variety of ways so that a wide swath of people will see it.

So here are a few tips used by some of the news outlets that have seen the most success with SecureDrop.

1.45.1 Make a high profile announcement

Anytime you launch a SecureDrop, you'll want to write an accompanying news story along with it to alert your readers and potential *Sources* where to submit information. Almost every news organization already does this, but some good recent examples come from USA Today, The Guardian, and Wired. You can also write a companion Q & A like the Washington Post did.

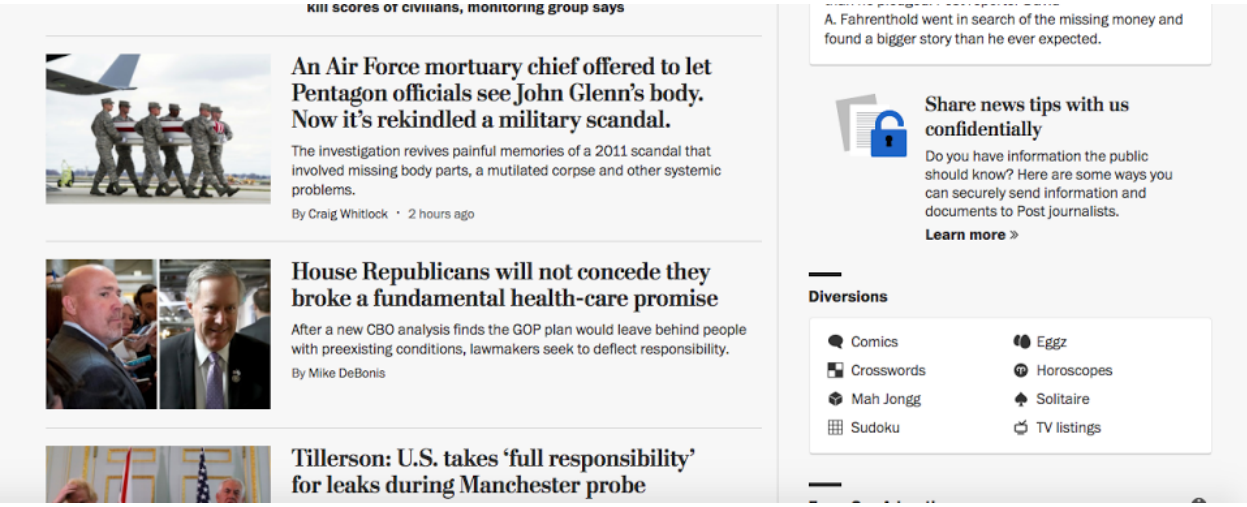
However, a launch announcement is really just a small piece of the puzzle. It's important to regularly remind readers and potential *Sources* that your SecureDrop exists, because only a tiny fraction will likely see the launch announcement and it will quickly be buried in other news after a couple of days.

1.45.2 Provide a clear link on your homepage

Making your SecureDrop or secure tips page easy to find is one of the most important things you can do to ensure that potential *Sources* use it. The best way you can do this is providing a clear link on your home page, so that every time a user goes to your website, they can quickly see where they need to go.

For example, the Intercept has a "become a source" link in its main menu:

The Washington Post has a link on their front page for "how to share a tip securely":



Other news organizations put a little link in their footer, however, we've found that this is not as effective as putting it in a more prominent on your front page.

1.45.3 Provide links at the bottom of your articles

Another great way to remind potential *Sources* know that they can use SecureDrop is to put a link at the bottom of each article. For example, Gizmodo Media Group, uses a message like this:

> Have something you think we should know? Email us at tips@deadspin.com, call our confidential tips hotline at **(347) 746-8471**, or contact our writers directly, or use our SecureDrop system. You can also follow us on Twitter, like us on Facebook, and sign up for our newsletter!

1.45.4 Create an instructional video on how to access and use your SecureDrop

To better help potential *Sources* visualize how SecureDrop works, several organizations have made short instructional videos walking through all the steps. Some good examples include the Toronto Globe and Mail, The Intercept, and Lucy Parsons Labs.

1.45.5 Regularly share your SecureDrop *Landing Page* on social media

The majority of adults in the United States now get their news from Facebook or other social media sites like Twitter, so it's important to regularly remind people via social media posts that SecureDrop is the safest way they can contact your *Journalists* if they have a sensitive tip to share. If there's specific stories you are looking for tips on that may already be in the news, this is a great way of getting added attention to your SecureDrop.

1.45.6 Target potential whistleblowers with advertising

Facebook and Twitter also allow for targeted advertising to users in specific locations, attributes, and sometimes even specific users. For example, Gizmodo Media Group targeted online advertisements for their secure tips page at DC residents imploring them to tell on trump. At Freedom of the Press Foundation, we ran a proof of concept Twitter advertisement aimed at EPA and NOAA employees to show how it can be done. You can read about how you can do the same thing here.

1.45.7 Put an advertisement in your physical paper

Obviously this tip only applies to news outlets that also print a physical newspaper, but putting an ad or in the paper to tell readers where to go to access SecureDrop can be extremely effective.

The New York Times took out a full page ad in their own paper when they launched SecureDrop and other secure communications tools for their tips line:


Runa Sandvik
@runasand
Following
So excited and proud to see @nytimes run a full page ad letting readers know how to securely send tips.
Got a confidential tip?
Get in touch.
Every news story starts somewhere. So if you think you have a credible tip, our newsroom is waiting to hear from you. Just keep in mind what we always keep in mind — is it a story of consequence? Will it impact society? And of course, is there documentation or evidence that can fuel the fact-based, original journalism that is a hallmark of The Times?
For secure, encrypted ways to share messages and materials with our journalists, visit nytimes.com/tips
RETWEETS
236
LIKES
544
7:45 AM - 17 Dec 2016
15
236
544

And the Toronto Globe and Mail regularly puts a note in their physical paper reminding potential *Sources* where they can go:

1.46 Using a YubiKey with the *Journalist Interface*

This guide describes in detail how to set up a YubiKey for two-factor authentication on the *Journalist Interface*. This setup is performed once per *Journalist* to create a secure log-in method. The process requires some configuration steps using a separate software tool.

Note

You will do all of these steps from within the Tails operating system.

1.46.1 What is a YubiKey?

A YubiKey is a physical token used for *Two-Factor Authentication*. They are made by a company called Yubico and are commercially available. Note that not all physical tokens are compatible with the YubiKey Personalization Tool; for this, you require a key that can support OATH-HOTP.

1.46.2 Download and launch the YubiKey personalization tool

1. Start Tails. At the log in-screen, choose the option to allow an administrator passphrase.
2. Open a terminal and enter

```
sudo apt-get update;
sudo apt-get install yubikey-personalization-gui
```

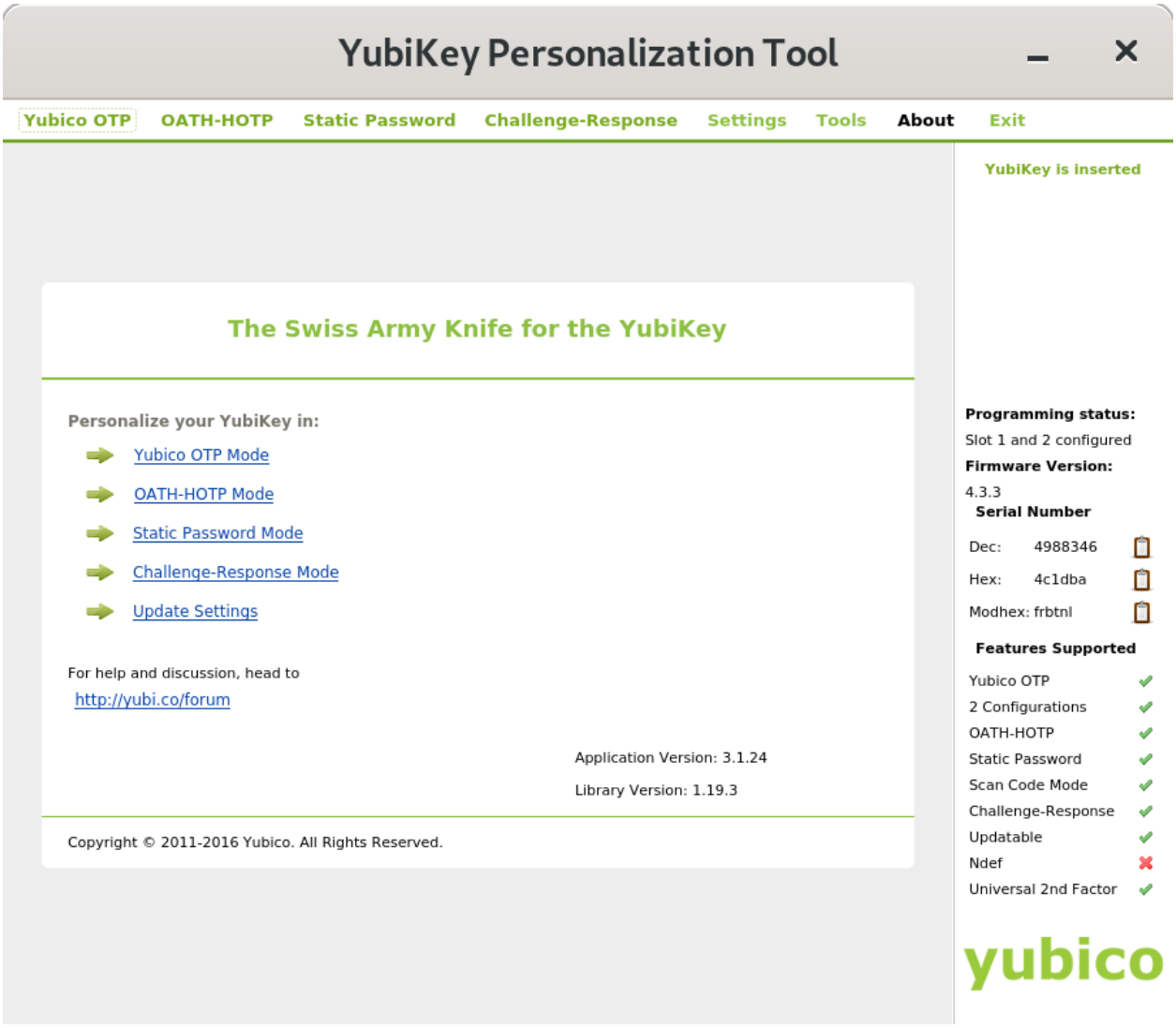
1. Once you have downloaded and installed the personalization program, open a **Root Console** by choosing **Apps ► System Tools ► Root Console**.
2. Open the YubiKey personalization tool by entering

```
yubikey-personalization-gui
```

1.46.3 Setting up hardware-based codes

After opening the personalization tool, click the heading **OATH-HOTP**. This will bring you to a window called **Program in OATH-HOTP mode**.

Click on the **Quick** button.



Under **Configuration Slot**, click **Configuration Slot 1**.

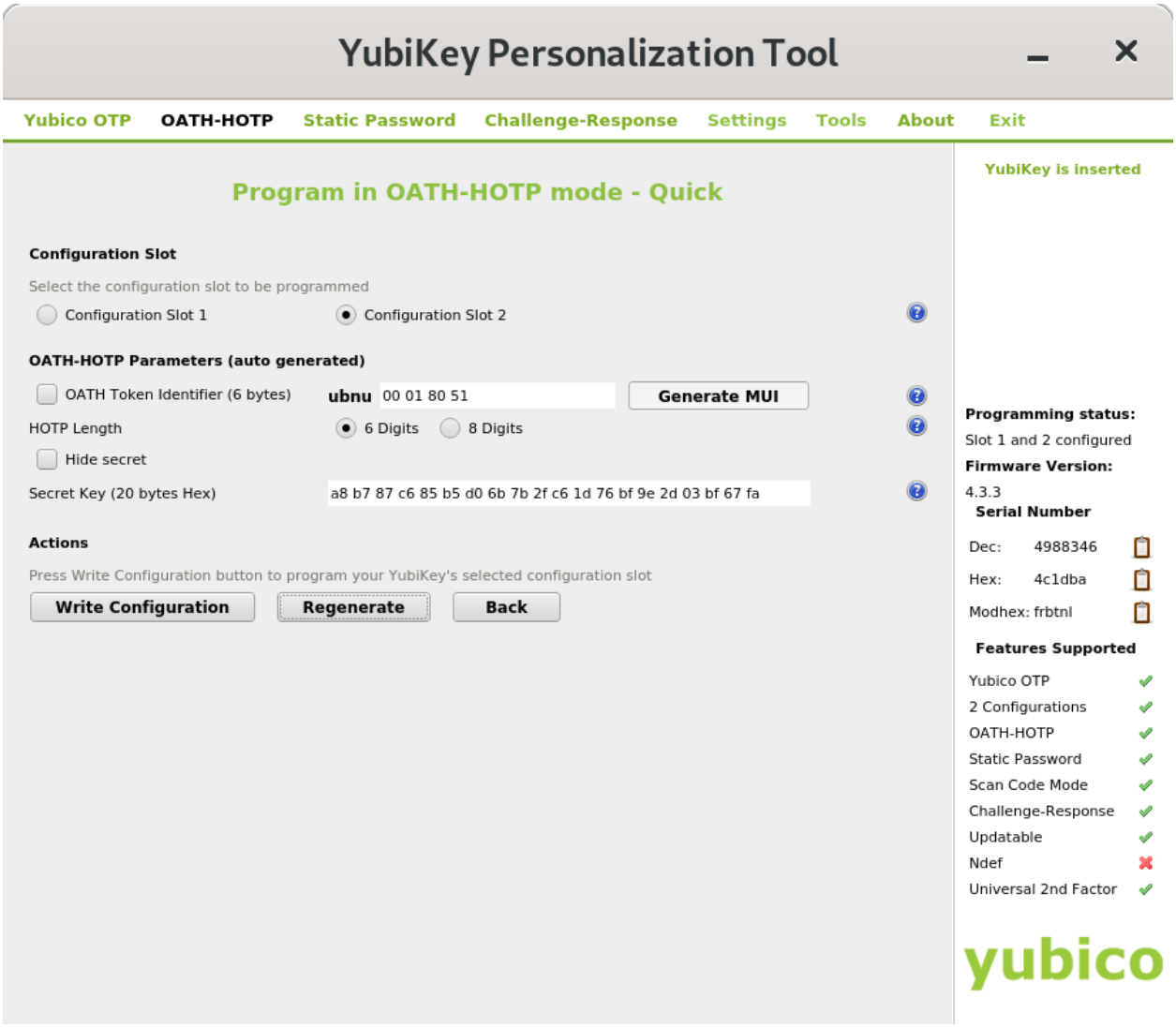
Note

If you are already using this YubiKey for something else, you should choose **Configuration Slot 2**. You will have to press and hold for several seconds to use the token from **Slot 2** instead of the one in **Slot 1**. See the YubiKey manual for more information.

In the section titled **OATH-HOTP parameters**, uncheck the box for **OATH Token Identifier (6 bytes)**. Leave the HOTP length at 6 digits. Next, uncheck the box for **Hide secret**. This will display the **Secret Key (20 bytes Hex)** field.

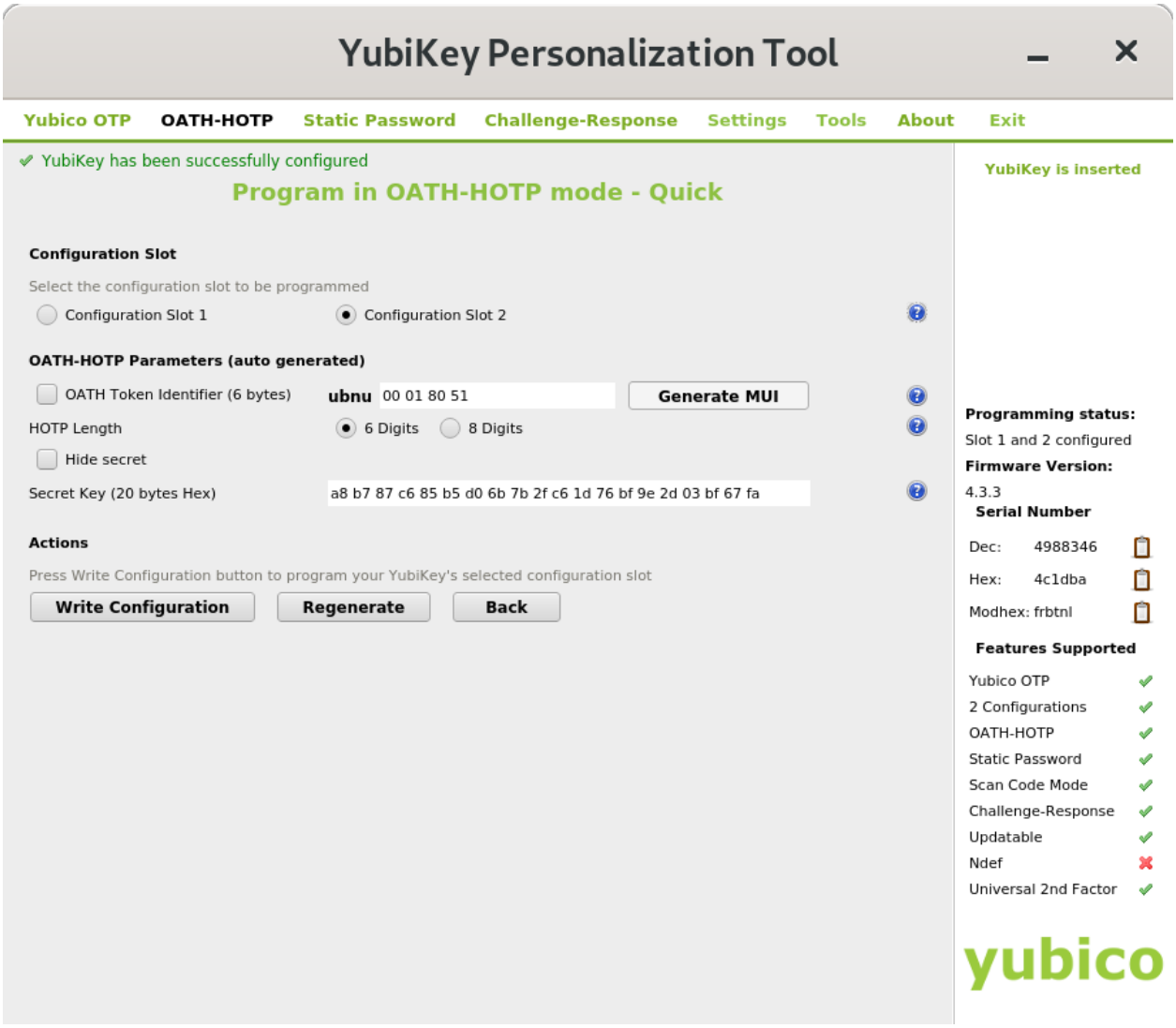
Important

Make a note somewhere safe of the **Secret Key (20 bytes Hex)** value.



When ready, click the **Write Configuration** button.

Click through the warning about overwriting the configuration slot and choose a location to save the log file. When the configuration is done, you should see green text saying **YubiKey configured** at the top of the window.



1.46.4 Adding users

When adding new users, a SecureDrop admin will need the **Secret Key** value described above. She will enter it after selecting the **I'm Using a YubiKey** option while *adding users*. The new user will then have to verify their YubiKey before being added to the system. This means that the new user and the admin should be physically present for this process.

1.46.5 Using your YubiKey

When using a Yubikey to log-in to the *Journalist Interface*, insert the Yubikey into the USB port and enter your username and passphrase. Then click the **Two-factor Code** field to focus the cursor there. Quickly press the lighted button on your YubiKey. This will insert the 6-digit code that you will need to log in.

Note

When using **Configuration Slot 2**, be sure to press and hold the YubiKey button for approximately 3 seconds. This can be somewhat finicky.

1.47 Tor proof-of-work defense on the *Source Interface*

The SecureDrop *Source Interface* is served as an *Onion Service* with an onion address, requiring Tor Browser to access it over the Tor network. Tor is sometimes targeted for denial-of-service (DoS) attacks that can slow down the Tor network as a whole as well as burden individual *Onion Services*, including SecureDrops.

Tor now includes a proof-of-work (PoW) defense against denial-of-service attacks that can be turned on for individual *Onion Services*. As of SecureDrop 2.9.0, new SecureDrops have this feature enabled by default, and we encourage all SecureDrop administrators to turn it on for their instances. While this measure can't speed up the Tor network as a whole if it's slow, it can protect your SecureDrop from being attacked specifically; and more *Onion Services* running with this feature helps improve the resilience of the Tor network.

1.47.1 Enabling the proof-of-work defense

If you're *installing SecureDrop for the first time*, the proof-of-work defense will be enabled by default, unless you *explicitly disable it*.

To enable it on an existing SecureDrop instance, on the *Admin VM*:

```
securedrop-admin sdconfig
```

The prompts will include:

```
Enable Tor's proof-of-work defense against denial-of-service attacks for the *Source␣
↪Interface*?: yes
```

Type <Enter> to accept the new default `yes` value. When you finish the prompts, rerun the installation script:

```
securedrop-admin install
```

The Tor configuration will be updated to enable the proof-of-work defense. When the script finishes, confirm that you can access the *Source Interface*.

1.47.2 Disabling the proof-of-work-defense

Follow the instructions above for *enabling the proof-of-work defense*, but answer `no` at the prompt:

```
Enable Tor's proof-of-work defense against denial-of-service attacks for the *Source␣
↪Interface*?: no
```

1.48 HTTPS on the *Source Interface*

The SecureDrop *Source Interface* is served as an *Onion Service* with an onion address ending in ".onion", requiring Tor Browser to access it. While *Onion Services* provide end-to-end encryption by default, as well as strong anonymity, there are several reasons why you might want to consider deploying an additional layer of encryption and authentication via HTTPS:

- Extended Validation (EV) certificates, which are currently the only type of certificates that may be issued for onion addresses, are intended to attest to the identity of the organization running a service. This provides an additional measure of authenticity (in addition to the organization's *Landing Page* and the SecureDrop Directory) to help assure *Sources* that they are communicating with the intended organization when they access a given *Source Interface*.
- Using HTTPS on the *Source Interface* will provide an extra layer of encryption for data in transit.

1.48.1 Obtaining an HTTPS certificate for onion addresses

Digicert

DigiCert is one of only two Certificate Authorities (CA) that issue HTTPS certificates for onion addresses. DigiCert requires organizations to follow the Extended Validation (EV) process in order to obtain a certificate for an Onion URL, so you should start by reviewing DigiCert's documentation for obtaining an HTTPS certificate for an onion address. The EV certificates display information about an organization under the certificate icon beside the URL bar:

Additional information about the organization, such as name and geographic location, are checked by the CA during the EV process. A *Source* can use this information to confirm the authenticity of a SecureDrop instance, beyond the verification already available in the SecureDrop Directory.

In order to obtain an HTTPS certificate for your SecureDrop instance, contact DigiCert directly. As part of the Extended Validation, you will be required both to confirm your affiliation with the organization, and to demonstrate control over the onion address for your *Source* Interface.

In order for you to demonstrate control over the onion address for your *Source* Interface, you will need to perform a signing operation leveraging the private key of the *Onion Service* used on the *Source* Interface. DigiCert will provide you with some text and request that you use that text in a signing operation. At a high level, obtaining a certificate from DigiCert involves:

1. Generating an HTTPS keypair and CSR via `openssl`.
2. Submitting the CSR to DigiCert. (This CSR demonstrates control over the private key used for HTTPS.)
3. Scheduling a phone call and verifying your relationship to the organization.
4. Generating another CSR, using a custom tool, leveraging the *Onion Service* private key.
5. Submitting the second CSR to DigiCert. (This CSR demonstrates control over the private key for the *Onion Service*.)
6. Downloading the certificate from the DigiCert panel.
7. Installing the cert on the SecureDrop *Application Server*, via `securedrop-admin`.

For SecureDrop, you should perform these steps on the *Admin Workstation*. Below are detailed steps for use on Tails:

```
# On the Admin Workstation, generate the first CSR
mkdir ~/Persistent/sd-https-key-generation
cd ~/Persistent/sd-https-key-generation
openssl req -new -newkey rsa:4096 -nodes -keyout sd.key -out sd.csr
```

That command will generate two files: `sd.key`, the private key that will be used by the SecureDrop *Application Server*; and `sd.csr`, the certificate signing request (CSR), that will be sent to certificate authority in order to receive a certificate. Upload that CSR to the DigiCert website, to begin the request. After passing the EV organization verification, you'll receive an email with a nonce. Use that value to generate the second CSR:

```
# On the Admin Workstation, generate the second CSR
source /usr/share/securedrop-admin/venv/bin/activate
torify pip install onionmaker
# Copy the *Onion Service* key material to the Admin Workstation:
mkdir hsdir
ssh app sudo cat /var/lib/tor/services/sourcev3/hostname > hsdir/hostname
```

(continues on next page)

(continued from previous page)

```
ssh app sudo cat /var/lib/tor/services/sourcev3/hs_ed25519_public_key > hsdir/hs_ed25519_
↪public_key
ssh app sudo cat /var/lib/tor/services/sourcev3/hs_ed25519_secret_key > hsdir/hs_ed25519_
↪secret_key
# Generate (second) CSR
onionmaker <nonce> hsdir
```

The CSR will be printed to stdout, starting with `BEGIN CERTIFICATE REQUEST`. Save that CSR, and send it via email reply to DigiCert. After you receive your final certificate, see instructions below for installing the certificate on the SecureDrop *Application Server*.

Harica

The Greek CA Harica is now providing Domain Validation (DV) certificates for onion addresses. DV certificates are less useful for authentication purposes, but may still be used to provide another layer of encryption for *Source* traffic. The commands provide detail on how to obtain a DV certificate from Harica on the *Admin Workstation*:

```
# On the Admin Workstation
cd ~/
git clone --recurse-submodules https://github.com/HARICA-official/onion-csr.git
cd onion-csr
sudo apt-get update && sudo apt-get install -y ruby-dev rubygems build-essential
# If prompted, choose to install the packages "Only once"
torify gem install --user-install ffi
gcc -shared -o libed25519.so -fPIC ed25519/src/*.c
# Confirm the binary works by checking that "help" info is displayed:
./onion-csr.rb -h

# Copy the Onion service key material to the Admin Workstation:
mkdir hsdir
ssh app sudo cat /var/lib/tor/services/sourcev3/hostname > hsdir/hostname
ssh app sudo cat /var/lib/tor/services/sourcev3/hs_ed25519_public_key > hsdir/hs_ed25519_
↪public_key
ssh app sudo cat /var/lib/tor/services/sourcev3/hs_ed25519_secret_key > hsdir/hs_ed25519_
↪secret_key

# Generate CSR
./onion-csr.rb -n <nonce> -d ./hsdir
```

1.48.2 Activating HTTPS in SecureDrop

Make sure you have *installed SecureDrop already*.

Make note of the *Source Interface* onion address. Now from a Terminal on your *Admin Workstation*:

```
securedrop-admin sdconfig
```

This command will prompt you for the following information:

```
Whether HTTPS should be enabled on *Source Interface* (requires EV cert): yes
Local filepath to HTTPS certificate (optional, only if using HTTPS on *Source␣
↪Interface*): sd.crt
Local filepath to HTTPS certificate key (optional, only if using HTTPS on *Source␣
```

(continues on next page)

(continued from previous page)

```
↪Interface*): sd.key
Local filepath to HTTPS certificate chain file (optional, only if using HTTPS on *Source␣
↪Interface*): ca.crt
```

The filenames should match the names of the files provided to you by DigiCert, and should be saved inside the `~/.config/securedrop-admin` directory. You'll rerun the configuration scripts:

```
securedrop-admin install
```

The webserver configuration will be updated to apply the HTTPS settings. Confirm that you can access the *Source Interface* at `https://<onion_address>.onion`, and also that the HTTP URL `http://<onion_address>.onion` redirects automatically to HTTPS.

Note

By default, Tor Browser will send an OCSP request to a Certificate Authority (CA) to check if the *Source Interface* certificate has been revoked. Fortunately, this occurs through Tor. However, this means that a CA or anyone along the path can learn the time that a Tor user visited the SecureDrop *Source Interface*. Future versions of SecureDrop will add OCSP stapling support to remove this request. See OCSP discussion for the full discussion.

1.49 SSH over local network

Under a production installation post-install, the default way to gain SSH administrative access is over the Tor network. This provides a number of benefits:

- Allows remote administration outside of the local network.
- Provides anonymity to an administrator while logging into the SecureDrop servers.
- Can mitigate against an attacker on your local network attempting to exploit vulnerabilities against the SSH daemon.

Most administrators will need SSH access during the course of running a SecureDrop instance and a few times a year for maintenance. So the potential shortfalls of having SSH over Tor are not usually a major issue. The cons of having SSH over Tor can include:

- Slow and delayed remote terminal performance
- Allowing SSH access from outside of your local network can be seen as a potential larger security hole for some organizations, particularly those with tight network security controls.

That being said, the default setting of only allowing SSH over Tor is a good fit for most organizations. If you happen to require SSH restricted to the local network instead please continue to read.

1.49.1 Configuring SSH for local access

Warning

It is important that your firewall is configured adequately if you decide you need SSH over the local network. The install process locks down access as much as possible with net restrictions, SSH keys, and *Two-Factor Authentication*. However, you could still leave the interface exposed to unintended users if you did not properly follow our network firewall guide.

Warning

This setting will lock you out of SSH access to your instance if your *Admin Workstation* passes through a NAT in order to get to the SecureDrop servers. If you are unsure whether this is the case, please consult your firewall configuration or network administrator.

Note

Whichever network you install from will be the one that SSH is restricted to post-install. This will come into play particularly if you have multiple network interfaces.

First, make sure your local SecureDrop environment is up-to-date and on the latest production release.

```
sudo apt update
securedrop-admin check_for_update
```

The setting that controls SSH over LAN access is set during the `sdconfig` step of the install. Below is an example of what the prompt will look like. You can answer either 'no' or 'false' when you are prompted for `Enable SSH over Tor`:

```
securedrop-admin sdconfig

Username for SSH access to the servers: vagrant
Local IPv4 address for the Application Server: 10.0.1.4
Local IPv4 address for the Monitor Server: 10.0.1.5
Hostname for Application Server: app
Hostname for Monitor Server: mon
[...]
Enable SSH over Tor (recommended, disables SSH over LAN). If you respond no, SSH will be␣
↪available over LAN only: no
```

Then you'll have to run the installation script:

```
securedrop-admin install
```

Note

If you are migrating from a production install previously configured with SSH over Tor, you will be prompted to re-run the `install` portion twice. This is due to the behind the scenes configuration changes being done to switch between Tor and the local network.

Finally, re-configure your *Admin Workstation* as follows:

```
securedrop-admin localconfig
```

Assuming everything is working you should be able to gain SSH access as follows:

```
ssh app
ssh mon
```

1.50 Configuring OSSEC fingerprint verification

If you run your own mail server, you may wish to increase the security level used by Postfix for sending mail to `fingerprint`, rather than `secure`. Doing so will require an exact match for the fingerprint of TLS certificate on the SMTP relay. The advantage to fingerprint verification is additional security, but the disadvantage is potential maintenance cost if the fingerprint changes often. If you manage the mail server and handle the certificate rotation, you should update the SecureDrop configuration whenever the certificate changes, so that OSSEC alerts continue to send. Using fingerprint verification does not work well for popular mail relays such as smtp.gmail.com, as those fingerprints can change frequently, due to load balancing or other factors.

You can retrieve the fingerprint of your SMTP relay by running the command below (all on one line). Please note that you will need to replace `smtp.gmail.com` and `587` with the correct domain and port for your SMTP relay.

```
openssl s_client -connect smtp.gmail.com:587 -starttls smtp < /dev/null 2>/dev/null |
    openssl x509 -fingerprint -noout -in /dev/stdin | cut -d'=' -f2
```

If you are using Tails, you will not be able to connect directly with `openssl s_client` due to the default firewall rules. To get around this, proxy the requests over Tor by adding `torify` at the beginning of the command. The output of the command above should look like the following:

```
6D:87:EE:CB:D0:37:2F:88:B8:29:06:FB:35:F4:65:00:7F:FD:84:29
```

Finally, add a new variable to `~/.config/securedrop-admin/site-specific` as `smtp_relay_fingerprint`, like so:

```
smtp_relay_fingerprint: "6D:87:EE:CB:D0:37:2F:88:B8:29:06:FB:35:F4:65:00:7F:FD:84:29"
```

Specifying the fingerprint will configure Postfix to use it for verification on the next playbook run. To apply the configuration, use `securedrop-admin` to *update the server configuration*.

To disable the fingerprint verification, simply delete the `smtp_relay_fingerprint` line and update the server configuration.

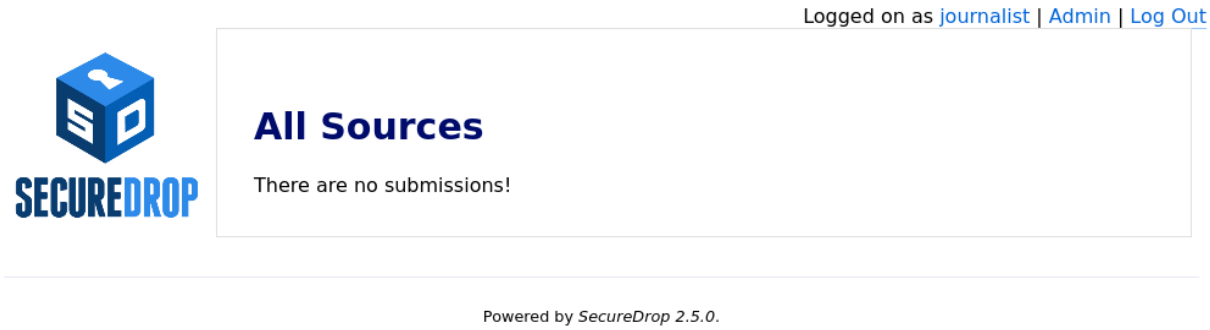
1.51 The *Admin Interface*

The *Admin Interface* is an extended version of the *Journalist Interface*, that allows you to manage users and configure the appearance and behaviour of your instance's web interfaces.

1.51.1 Logging in

To log in to the *Admin Interface*, start the *Admin Workstation* with persistence enabled. Open the *SecureDrop Menu* and select the **Launch Journalist Interface** option. Tor Browser will start and load the login page for the *Journalist Interface*. Use your username, passphrase, and *Two-Factor Authentication* token to log in.

By default, you will be logged in to the *Journalist Interface*'s source list page.



In the course of normal administration operations you should not need to view messages from *Sources*, but if you do, you can find information on managing submissions in the *journalist guide*.

Note

If you have lost your login information or your *Two-Factor Authentication* is no longer valid, you can create another account with admin privileges via the command line on the *Application Server*. See *here* for more information.

1.51.2 User management

You can use the *Admin Interface* to add and remove users, and to reset their credentials if necessary. To open the *Admin Interface*, click **Admin** in the upper right corner of the *Journalist Interface*.

Adding users

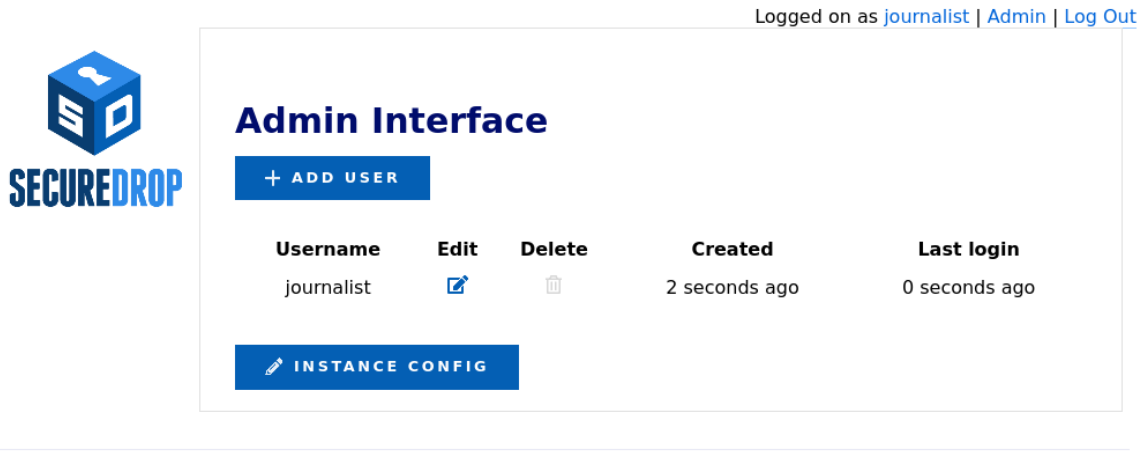
After logging in, you can add new user accounts for the *Journalists* at your organization who will be checking the system for submissions. Make sure the *Journalist* is physically in the same room as you when you do this, as they will have to be present to enable *Two-Factor Authentication*. SecureDrop supports the use of either a smartphone authenticator app or a Yubikey for *Two-Factor Authentication*. If an app is to be used, the *Journalist* should install it before proceeding with the account setup.

Tip

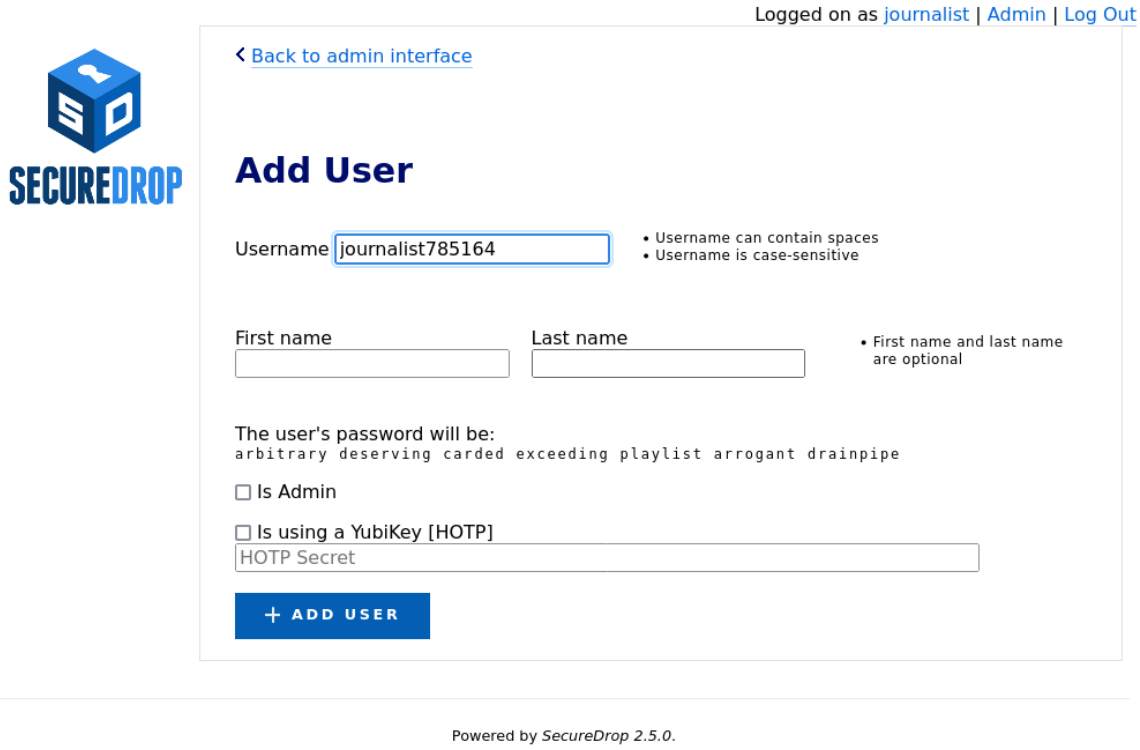
We recommend using FreeOTP (available for Android and for iOS) to generate two-factor codes because it is Free Software. However, if it does not work for you for any reason, alternatives exist:

- Google Authenticator for Android and iOS (proprietary)
- authenticator for the desktop (Free Software)

1. Click **Admin** in the top right corner of the page to load the *Admin Interface*.



2. Click **Add User** to add a new user.



3. Hand the keyboard over to the *Journalist* so they can create their own username.
4. Once they're done entering a username for themselves, have them save their pre-generated Diceware passphrase to their password manager.
5. If the new account should also have admin privileges, allowing them to add or delete other journalist accounts, select **Is Admin**.
6. Finally, set up *Two-Factor Authentication* for the account, following one of the two procedures below for your chosen method.

Note

The username **deleted** is reserved, as it is used to mark accounts which have been deleted from the system.

FreeOTP

1. If the *Journalist* is using FreeOTP or another app for *Two-Factor Authentication*, click **Add User** to proceed to the next page.

2. Next, the *Journalist* should open FreeOTP on their smartphone and scan the barcode displayed on the screen.
3. If they have difficulty scanning the barcode, they can tap on the icon at the top that shows a plus and the symbol of a key and use their phone's keyboard to input the two-factor secret into the `Secret` input field, without whitespace.
4. Inside the FreeOTP app, a new entry for this account will appear on the main screen, with a six-digit number that recycles to a new number every thirty seconds. The *Journalist* should enter the six-digit number in the **Verification code** field at the bottom of the **Enable FreeOTP** form and click **Submit**.

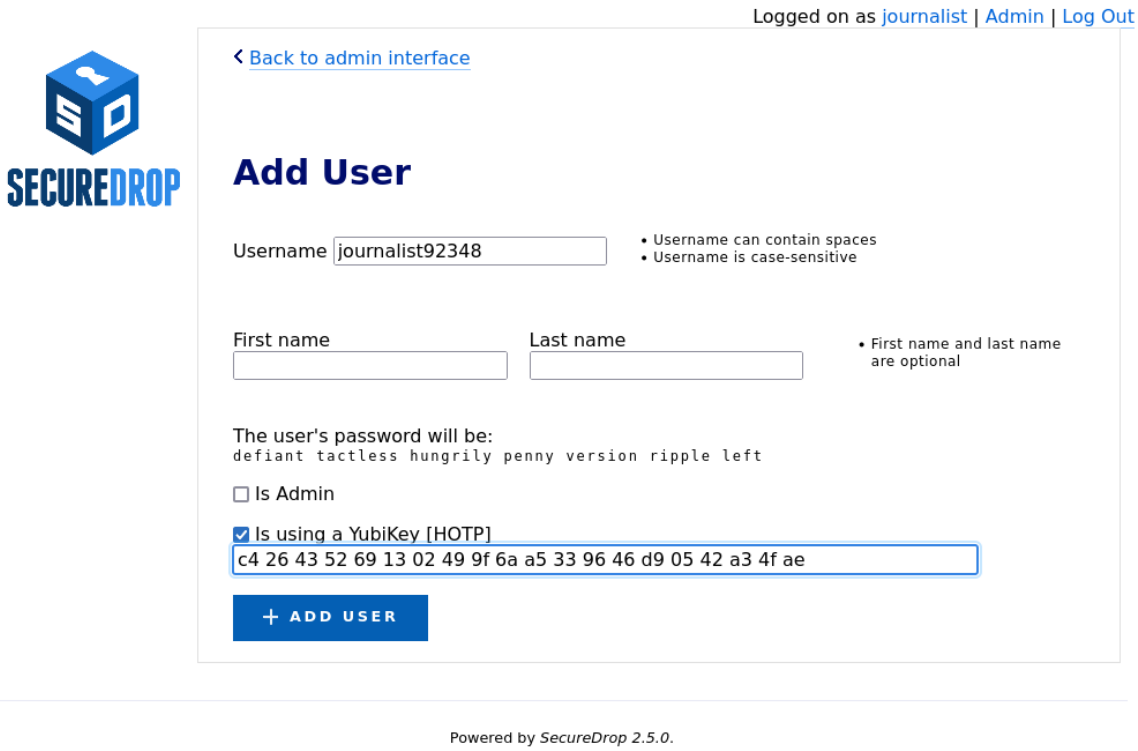
If *Two-Factor Authentication* was set up successfully, you will be redirected back to the *Admin Interface* and will see a confirmation that the two-factor code was verified.

Note

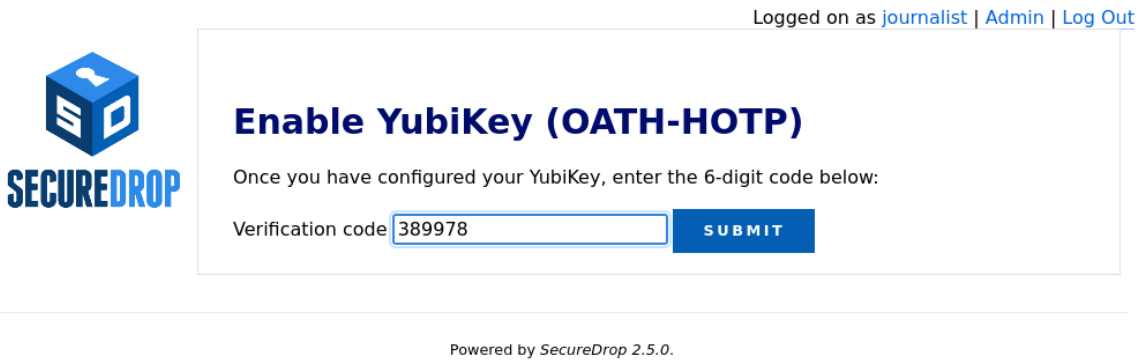
If the QR code for setting up *Two-Factor Authentication* in your mobile authenticator app is not displayed, it may be blocked by Tor Browser. You can set Tor Browser's security level to **Standard** by clicking on the Shield icon. Alternatively, you can manually type in the two-factor secret (in FreeOTP, use the **Add token** option from the menu).

YubiKey

1. If the *Journalist* wishes to use a YubiKey for *Two-Factor Authentication*, select **Is using a YubiKey**. You will then need to enter their YubiKey's OATH-HOTP Secret Key. For more information on how to retrieve this key, read the *YubiKey Setup Guide*.



2. Once you've entered the Yubikey's OATH-HOTP Secret Key, click **Add User**. On the next page, have the *Journalist* authenticate using their YubiKey, by inserting it into a USB port on the workstation and pressing its button.



3. If everything was set up correctly, you will be redirected back to the *Admin Interface*, where you should see a flashed message that says "The two-factor code for user *new username* was verified successfully.".

The *Journalist* will require their username, passphrase, and *Two-Factor Authentication* method whenever they check SecureDrop. Make sure that they have memorised their username and passphrase, or stored them in their password manager, and that they can keep their *Two-Factor Authentication* device secure.

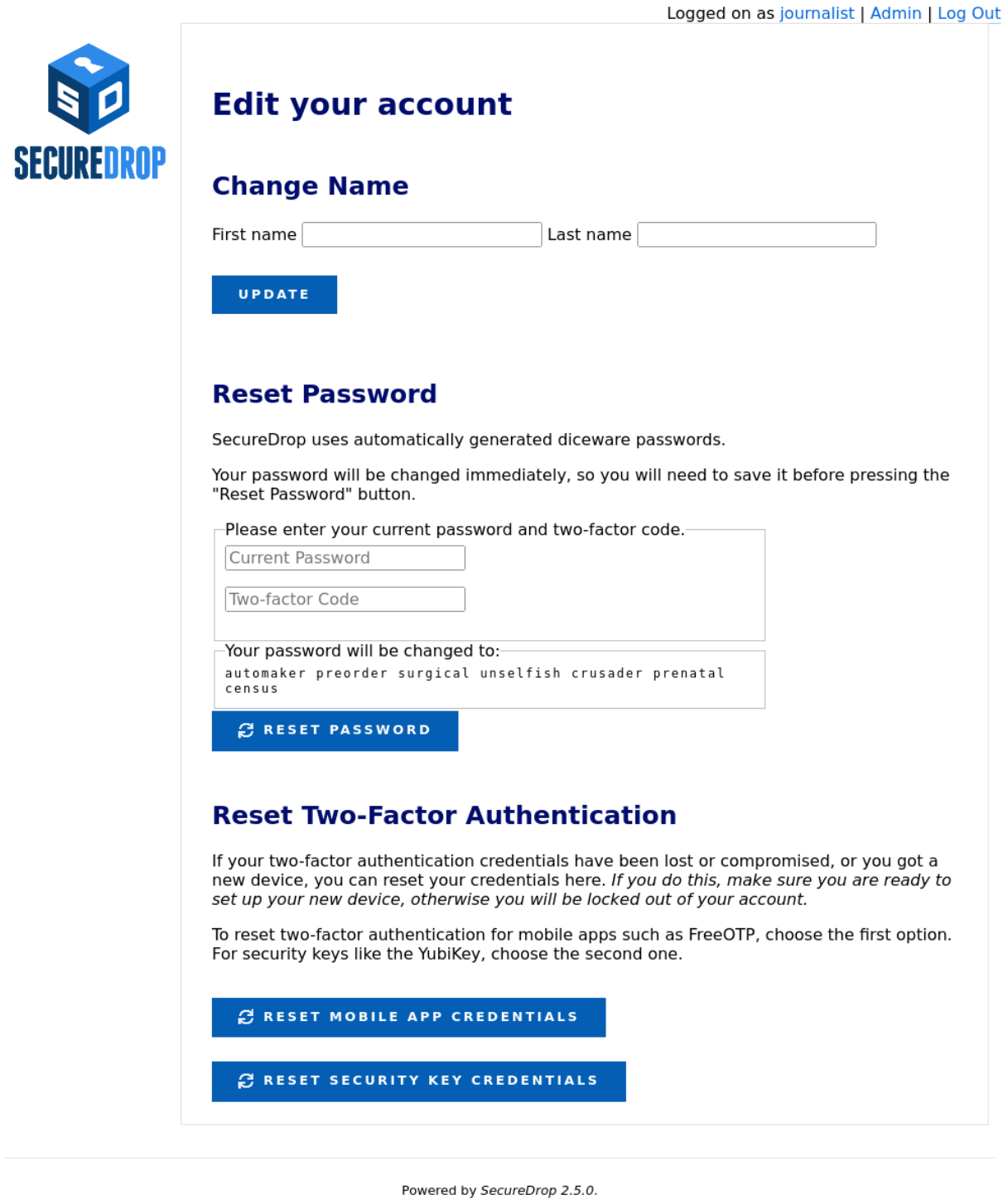
Passphrases and *Two-Factor Authentication* resets

Warning

Both of these operations will lock a user out of their SecureDrop account. Users should be physically present when their passphrase or *Two-Factor Authentication* method is reset. If this is not possible, store the passphrase and/or *Two-Factor Authentication* secret in your own password manager before securely transmitting them to the user in question, and delete them once the user has confirmed they can successfully log in.

Even while following *passphrase best practices*, your *Journalists* may occasionally lock themselves out of their accounts. This can happen if, for example, they lose their two-factor device or if they forget the passphrase to their password manager. When this happens, you can reset their account as follows:

1. Log in as an administrator to the *Journalist Interface*
2. Select *Admin* at the top right to open the *Admin Interface*
3. Find the user's account name and select **Edit**



Next, you can either rotate their passphrase or reset *Two-Factor Authentication* for their account.

To change their passphrase to the randomly-generated passphrase shown:

1. Have the *Journalist* enter their current passphrase and two-factor code.
2. Make sure the new passphrase is saved in a password manager.
3. Click **Reset Password**

To reset *Two-Factor Authentication*:

1. Click the button that corresponds to the user's chosen *Two-Factor Authentication* method:
   - Click **Reset Mobile App Credentials** for accounts using FreeOTP or a similar authentication app
   - Click **Reset Security Key Credentials** for accounts using a Yubikey
2. Follow the on-screen instructions to complete the process and verify their new *Two-Factor Authentication* credentials.

Off-boarding users

See *our guide to off-boarding users from SecureDrop*.

1.51.3 Instance configuration

The Instance Configuration section of the *Admin Interface* allows you to:

- update the organization name and logo displayed on the *Source* and *Journalist Interfaces*
- set submission preferences for the *Source Interface*
- send test OSSEC alerts.

Updating the organization name

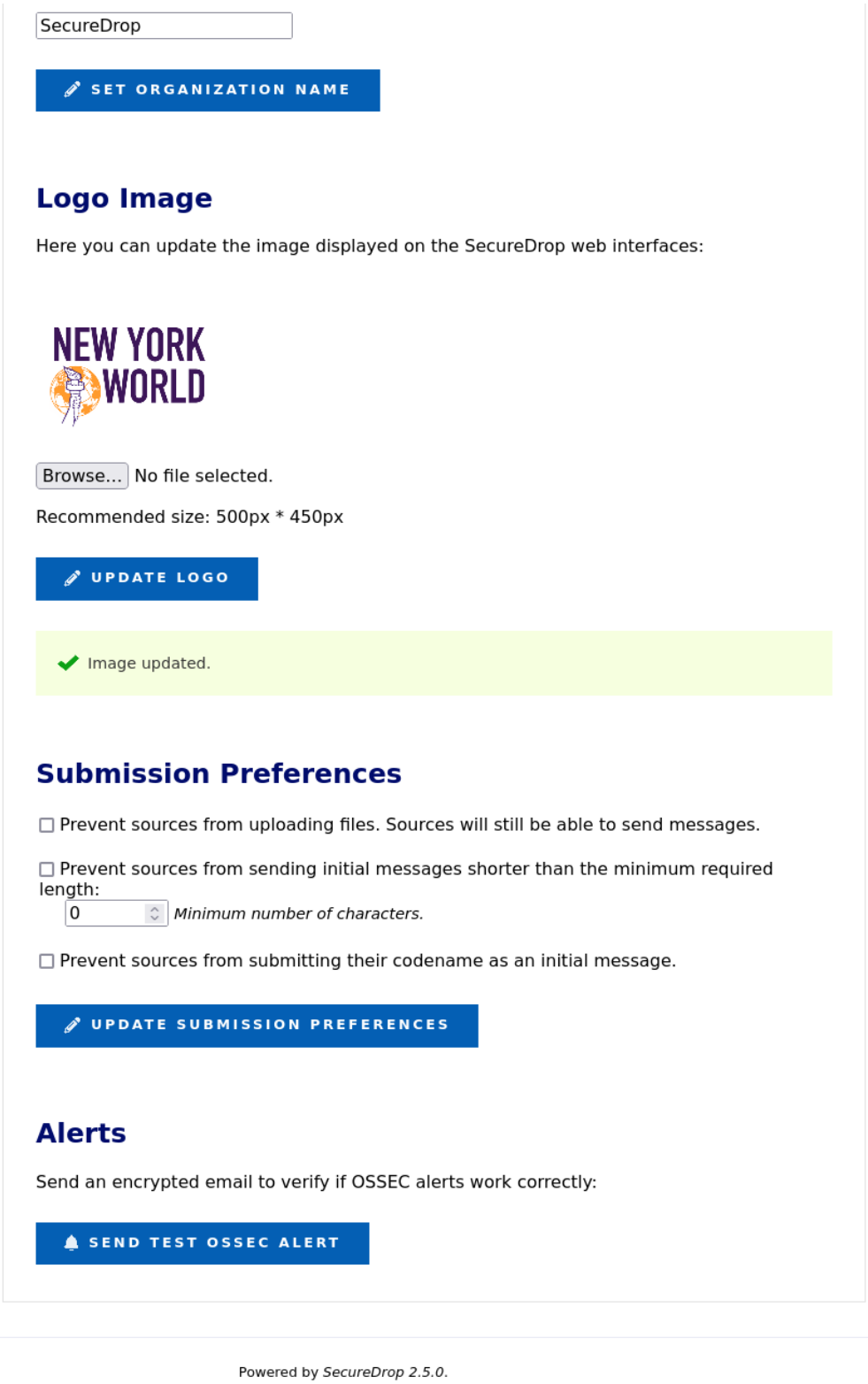
Your organization name is used in page titles and logo ALT text on the *Source Interface* and *Journalist Interface*. By default, it's set to `SecureDrop`. To change it, enter your desired name in the Organization Name field and click **Set Organization Name**.

Updating the logo image

You can update the system logo shown on the web interfaces of your SecureDrop instance via the *Admin Interface*. We recommend a size of `500px x 450px`. Only PNG-format images are supported. To update the logo image:

1. Copy the logo image to your *Admin Workstation*
2. Click **Browse** and select the image from your workstation's filesystem
3. Click **Update Logo** to upload and set the new logo

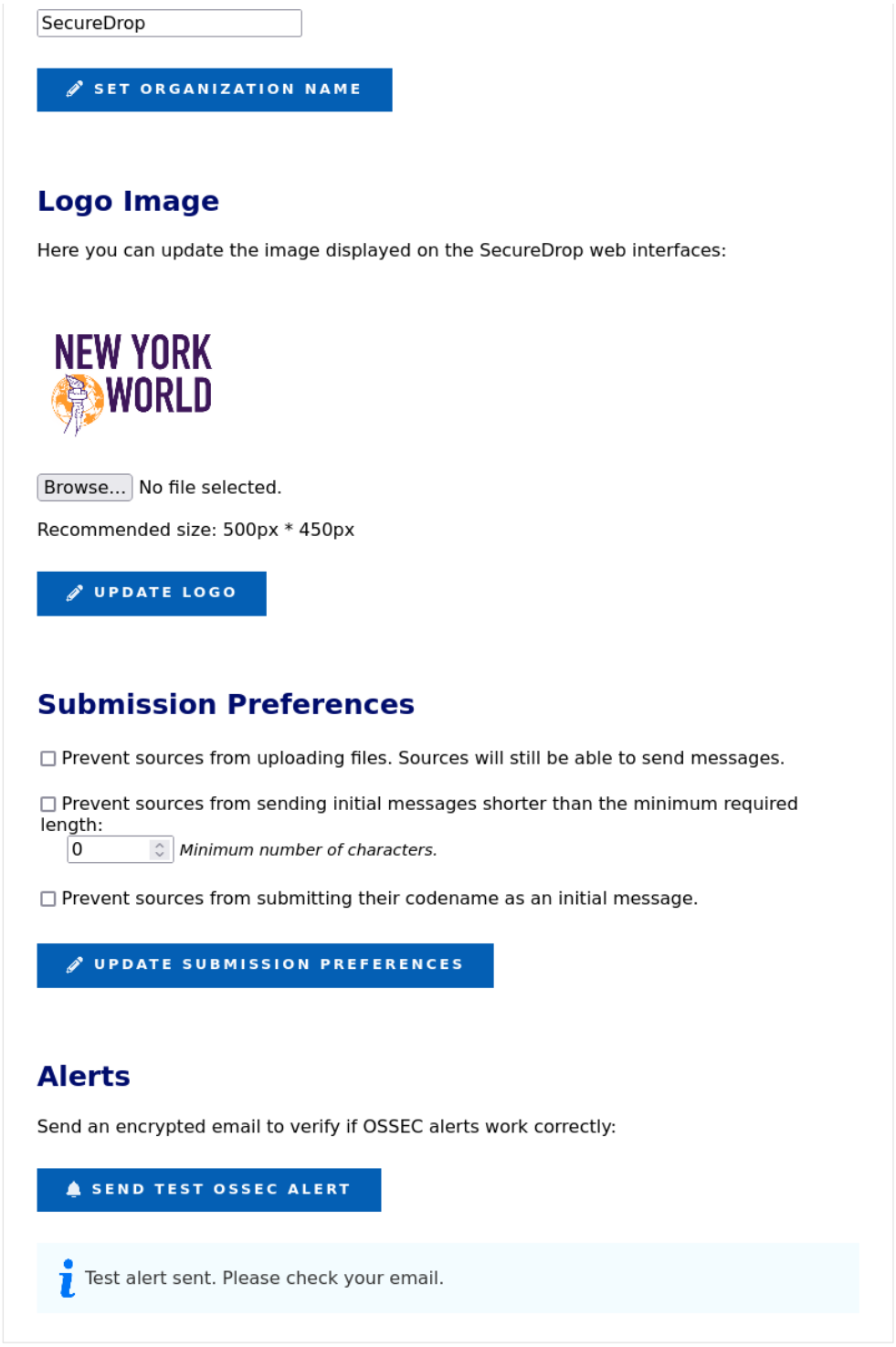
You should see a message appear indicating the change was a success.



It may be necessary to hold the Shift key while pressing the Reload button in the browser, which will force it to purge the cached version of the logo in order to see the new one.

Testing OSSEC alerts

To verify that the OSSEC monitoring system's functionality, you can send a test OSSEC alert by clicking **Send Test OSSEC Alert**.



You should receive an OSSEC alert email at the address specified during the installation of SecureDrop. The email may take several minutes to arrive. If you don't receive it, refer to the *OSSEC Guide* for information on troubleshooting steps.

Submission preferences

The Submission Preferences subsection allows you to restrict the types of submissions accepted by your instance.

Disabling document uploads

By default, SecureDrop supports both text submissions and document uploads. If you only want to receive text messages, you can disable uploads as follows:

1. Check the **Prevent sources from uploading documents** checkbox
2. Click **Update Submission Preferences**

This change will be applied immediately on the *Source Interface*. Documents that were previously uploaded will still be available via the *Journalist Interface*.

Preventing short initial messages

By default, SecureDrop does not apply a minimum length requirement to messages. If your instance is experiencing a high volume of short one-time messages with no actionable content, or if you would like to indicate to *Sources* that their initial message should include enough information for *Journalists* to respond to them effectively, you can set an initial message length as follows:

1. Check the **Prevent sources from sending initial messages shorter than the minimum required length** checkbox
2. Enter the desired minimum length in the field below the checkbox
3. Click **Update Submission Preferences**

This change will be applied immediately on the *Source Interface*. Initial messages that are too short will be rejected, with an error message informing *Sources* of the requirement. This requirement will not be applied to initial messages that also include a document, or to subsequent messages in the conversation.

To remove the requirement, uncheck the checkbox and click **Update Submission Preferences**.

Preventing initial messages containing the *Source*'s codename

Sources should never need to share their seven-word codename with *Journalists*. If your instance is receiving one-time messages consisting of the *Source*'s codename, you can optionally reject those messages, before they are stored, as follows:

1. Check the **Prevent sources from submitting their codename as an initial message** checkbox
2. Click **Update Submission Preferences**

This change will be applied immediately on the *Source Interface*. Initial messages that contain the *Source*'s codename will be rejected, with an error message reminding *Sources* to protect their codename and keep it secret. To remove this restriction, uncheck the checkbox and click **Update Submission Preferences**.

1.52 Analyzing the alerts

Understanding the contents of the OSSEC alerts requires a background and knowledge in Linux systems administration. They may be confusing, and at first it will be hard to tell between a genuine problem and a fluke. You should examine these alerts regularly to ensure that the SecureDrop environment has not been compromised in any way, and follow up on any particularly concerning messages with direct investigation.

An initial SecureDrop install will generate quite a few alerts as OSSEC is installed early in the install process. As part of the administration of a SecureDrop instance, regularly looking through the generated alerts provides administrators with information on the overall health of the SecureDrop instance.

OSSEC alerts will range from a severity level of 1 (lowest) to 14 (highest), and as a baseline, you should expect to see the following alerts:

1.52.1 Common OSSEC alerts

Package updates

The SecureDrop *Application* and *Monitor Servers* check for package updates every day. As updates are automatically installed, OSSEC will notice and send out alerts. You may see any number of these alerts in the email, as several alerts can be batched in a single email. You should also see them in an email named `Daily Report: File Changes`. To verify this activity matches the package history, you can review the logs in `/var/log/apt/history.log`.

```
Received From: (app)
Rule: 2902 fired (level 7) -> "New dpkg (Debian Package) installed."
Portion of the log(s):

status installed <package name> <version>
```

In addition to letting you know what packages were updated, OSSEC will send alerts about the specific changes to the files in these packages.

```
Received From: (app)
Rule: 550 fired (level 7) -> "Integrity checksum changed."
Portion of the log(s):

Integrity checksum changed for: '/usr/sbin/<binary name>'
Old md5sum was: '<old md5sum>'
New md5sum is : '<new md5sum>'
Old sha1sum was: '<old sha1sum>'
New sha1sum is : '<new sha1sum>'
```

It may seem redundant to receive both `New dpkg (Debian Package) installed` and `Integrity checksum changed` alerts. This happens because OSSEC's alerts do not track root causation: OSSEC doesn't "know" that files have changed because new packages have been installed or updated, so it reports both sets of events independently. As a result, these clusters of alerts are normal and expected: they tell you that your SecureDrop servers are properly up-to-date and patched.

Keep an eye out for *exceptions* to this rule as you analyze your OSSEC alerts. Surprising changes to configuration files, or new or changed files unrelated to the daily updates, may warrant further investigation.

Occasionally your SecureDrop Servers will send an alert for failing to connect to Tor relays. Since SecureDrop runs as a Tor *Onion Service*, it is possible for Tor connections to timeout or become overloaded.

```
Received From: (app)
Rule: 1002 fired (level 2) -> "Unknown problem somewhere in the system."
Portion of the log(s):

[warn] Your Guard <name> ($fingerprint) is failing a very large amount of
circuits. Most likely this means the Tor network is overloaded, but it
could also mean an attack against you or potentially the guard itself.
```

This alert is common but if you see them for sustained periods of time (several times a day), please *contact us* for help.

Daily reports

On days where file integrity checksums have changed or users have logged into `app` or `mon` servers, you will receive emails entitled `Daily report: File changes` or `Daily report: Successful logins`. These emails may be a more convenient format should you not have continuous access to the inbox or GPG key.

Action: periodically review these daily reports to ensure file changes correspond to platform updates and logins correspond to authorized admin activity on the SecureDrop servers.

If you have any suggestions on how to further tune or improve the alerting, you can open an issue on GitHub.

1.52.2 Uncommon OSSEC alerts

Data integrity

SecureDrop runs automatic checks for submission data integrity problems. For example, secure deletion of large submissions could potentially be interrupted:

```
Received From: (app) 10.20.2.2->/opt/venvs/securedrop-app-code/bin/python3 /var/www/
↪securedrop/manage.py check-disconnected-fs-submissions
Rule: 400801 fired (level 1) -> "Indicates that there are files in the submission area␣
↪without corresponding submissions in the database."
Portion of the log(s):

ossec: output: '/opt/venvs/securedrop-app-code/bin/python3 /var/www/securedrop/manage.py␣
↪check-disconnected-fs-submissions': There are files in the submission area with no␣
↪corresponding records in the database. Run "manage.py list-disconnected-fs-submissions
↪" for details.
```

To resolve the issue, you can *clean them up*.

Instance misconfigurations

In addition, SecureDrop performs a small set of daily configuration checks to ensure that the iptables rules configured on the *Application* and *Monitor Server* match the expected configuration. If they do not, you may receive a level 12 alert like the following:

```
Received From: (app) 10.20.2.2->/var/ossec/checksdconfig.py
Rule: 400900 fired (level 12) ->
"Indicates a problem with the configuration of the SecureDrop servers."
Portion of the log(s):
ossec: output: '/var/ossec/checksdconfig.py': System configuration error:
The iptables default drop rules are incorrect.
```

Alternatively, the error text may say: `The iptables rules have not been configured.` To resolve the issue, you can reinstate the standard iptables rules by *updating the system configuration*.

`securedrop-admin` commands

OSSEC will send an alert when the *securedrop-admin* tool is used to backup, restore, or change the system configuration:

```
Rule: 400001 fired (level 13) -> "Ansible playbook run on server (securedrop-admin␣
↪install, backup, or restore)."
```

Action: You should ensure that this action was performed by you or a fellow administrator.

If you believe that the system is behaving abnormally, you should *contact us* for help.

1.53 Logging in via SSH

1.53.1 SSH over Tor

By default, SSH access to SecureDrop servers is routed through the Tor network, allowing you to access the servers from anywhere in the world where you have a stable internet connection and are able to access the Tor network.

To do so, simply open a Terminal and run either the `ssh app` or `ssh mon` command, depending on which server you are intending to access.

This is useful for routine maintenance and log investigation tasks, although direct physical access will still be necessary for network-related issues, in situations where SSH access is not available.

For more details about the types of tasks that can be completed via SSH, you can *review the SSH portion of our Admin Guide*.

If you'd like to make adjustments to the SSH configuration, or disable SSH access over Tor, you can do so by following the steps here.

In addition to remote SSH access, the web-based *Admin Interface* is also from any location with a network connection and access to the Tor network.

1.53.2 Server SSH access

Generally, you should avoid directly SSHing into the servers in favor of using the *Admin Interface* or `securedrop-admin`. However, in some cases, you may need to SSH in order to troubleshoot and fix a problem that cannot be resolved via these tools.

You can access your *Application Server* and *Monitor Server* via SSH by using either the `ssh app` or `ssh mon` commands (respectively).

In this section we cover basic commands you may find useful when you SSH into the *Application Server* and *Monitor Server*.

Tip

When you SSH into either SecureDrop server, you will be dropped into a `tmux` session. `tmux` is a screen multiplexer - it allows you to tile panes, preserve sessions to keep your session alive if the network connection fails, and more. Check out this tmux tutorial to learn how to use `tmux`.

Tip

If you want a refresher of the Linux command line, we recommend this resource to cover the fundamentals.

Shutting Down the Servers

```
sudo shutdown now -h
```

Rebooting the servers

```
sudo reboot
```

1.53.3 Investigating logs

Consult our *Investigating Logs* topic guide for locations of the most relevant log files you may want to examine as part of troubleshooting, and for how to enable error logging for the *Source Interface*.

Note

You can use the `securedrop-admin` tool to extract logs to send to Freedom of the Press Foundation for analysis. Run the following command:

```
securedrop-admin logs
```

This command will produce encrypted tarballs containing logs from each server. See the command output for more information.

1.53.4 Immediately apply a SecureDrop update

SecureDrop will update and reboot once per day. However, once a SecureDrop update is announced, you can opt to fetch the update immediately.

Important

Except where otherwise indicated, make sure to update both your *Application Server* and your *Monitor Server*.

To update your servers immediately, you can SSH into each server (via `ssh app` and `ssh mon`) and run the following commands:

```
sudo apt update
sudo unattended-upgrades
```

Note

Depending on the nature of the update (e.g., if the `tor` package is upgraded and you are using SSH-over-Tor), your SSH connection may be interrupted, and you may have to reconnect to see the full output.

1.53.5 *Application Server*

Adding users (CLI)

After the provisioning of the first admin account, we recommend using the *Admin Interface* web application for adding additional journalist and admin accounts.

However, you can also add users via `./manage.py` in `/var/www/securedrop/` as described *during first install*. You can use this command line method if the web application is unavailable.

Restart the web server

If you make changes to your Apache configuration, you may want to restart the web server to apply the changes:

```
sudo systemctl restart apache2
```

Cleaning up deleted submissions

When submissions are deleted through the web interface, their database records are deleted and their encrypted files are securely wiped. For large files, secure removal can take some time, and it's possible, though unlikely, that it can be interrupted, for example by a server reboot. In older versions of SecureDrop this could leave a submission file present without a database record.

As of SecureDrop 1.0.0, automated checks send OSSEC alerts when this situation is detected, recommending you run `manage.py list-disconnected-fs-submissions` to see the files affected. As with any `manage.py` usage, you would run the following:

```
ssh app
sudo -u www-data bash
cd /var/www/securedrop
./manage.py list-disconnected-fs-submissions
```

You then have the option of running:

```
./manage.py delete-disconnected-fs-submissions
```

to clean them up. As with any potentially destructive operation, it's recommended that you *back the system up* before doing so.

There is also the inverse scenario, where a database record could point to a file that no longer exists. This would usually only have happened as a result of disaster recovery, where perhaps the database was recovered from a failed hard drive, but some submissions could not be. The OSSEC alert in this case would recommend running:

```
./manage.py list-disconnected-db-submissions
```

To clean up the affected records you would run (again, preferably after a backup):

```
./manage.py delete-disconnected-db-submissions
```

Even when submissions are completely removed from the application server, their encrypted files may still exist in backups. We recommend that you delete old backup files with `shred`, which is available on Tails.

1.53.6 *Monitor Server*

Restart OSSEC

If you make changes to your OSSEC monitoring configuration, you will want to restart OSSEC via OSSEC's control script, `ossec-control`:

```
sudo /var/ossec/bin/ossec-control restart
```

1.54 Off-board administrators and *Journalists*

When *Journalists* and SecureDrop administrators leave your organization, it is important to off-board them from SecureDrop.

Important

Additional measures may need to be taken if the user's departure is on unfriendly terms. These measures will vary depending on the circumstances and your own internal incident response procedures, and may include doing a full reinstall of SecureDrop. If you are in such a situation, feel free to *contact us* for further assistance.

1.54.1 Off-boarding checklist

- *Inform the SecureDrop Support* team that the user should be removed from any support Signal groups, and indicate if any new staff members should be added.
- Delete the user's account on the *Journalist Interface*.
- Retrieve *SecureDrop Workstation* laptops, *Backup* drive(s), and any other SecureDrop hardware or materials.
- If the user receives email alerts (OSSEC alerts or daily submission notifications), either directly or as a member of an email alias, remove them from those alerts and *set up someone new* to receive those alerts.
- (Circumstance-dependent) If you have specific concerns that the *Submission Private Key* has been compromised, you should consider a full reinstall of SecureDrop. At minimum, you should *rotate the *Submission Key**.

1.54.2 Additional steps for off-boarding administrators

- If the departing user was your primary SecureDrop admin, designate the next person who will take over their function. Ideally, your outgoing administrator will be able to provide as much training as possible on the use and maintenance of the system, as well as on your organizational policies (such as backup strategies, and so on) before they leave; if this is not the case, *contact the SecureDrop Support team*.
- We do not recommend enabling remote management for SecureDrop's network firewall. However, if your SecureDrop firewall can be accessed remotely, even if only from within your organization's network, you may want to rotate its login credentials.
- Back up and *rotate the SSH key* to prevent unauthorized SSH access to the *Application* and *Monitor Servers* in the event that this user has retained their Admin SSH credentials.

Rotate SSH keys on the SecureDrop servers

If you are concerned that the user may have a copy of the SSH key, you should rotate the key in the following manner.

1. Create a new SSH keypair. From an *Admin* VM, run

   ```
   ssh-keygen -t rsa -b 4096
   ```

   and make sure to change the key name. This is the only parameter you need to change. For example, instead of `/home/amnesia/.ssh/id_rsa`, call the key `/home/amnesia/.ssh/newkey`. You don't need a passphrase for the key.

2. Copy new public key to the SecureDrop Servers. Copy the public portion of the key to the *Application* and *Monitor Servers* by running

   ```
   scp -O /home/amnesia/.ssh/newkey.pub scp://app
   ```

   and

   ```
   scp -O /home/amnesia/.ssh/newkey.pub scp://mon
   ```

3. Add this key to the list of authorized keys. SSH to the *Application Server* and append this new key to the list of authorized keys by using

   ```
   cat newkey.pub >> ~/.ssh/authorized_keys
   ```

   Be sure to use the command as above so that you append the key, instead of replacing the file. While you are still on the *Application Server*, you can then delete the file `newkey.pub` from wherever you scp'd it to (i.e. your home directory). Repeat this process with the *Monitor Server*.

4. Rename SSH keys. Exit all SSH sessions and rename `id_rsa` and `id_rsa.pub` (the old SSH keys) to something else. For example,

```
mv /home/amnesia/.ssh/id_rsa /home/amnesia/.ssh/id_rsa_old
mv /home/amnesia/.ssh/id_rsa.pub /home/amnesia/.ssh/id_rsa_old.pub
```

   Then, rename your `newkey` and `newkey.pub` to `id_rsa` and `id_rsa.pub`.

5. Test SSH connection. Test that you can still ssh into the *Application* and *Monitor Servers* (you can test with `ssh app host` and `ssh mon host`).

6. Restrict SSH access to the new key.

> **Important**
>
> If you have other users who also have SSH access to the *Application* and *Monitor Servers*, the next step will revoke their access. Their public keys will have to be re-appended to the `authorized_keys` file on each server, as in step 3.

   From an *Admin VM*, run

```
securedrop-admin reset_admin_access
```

   This removes all other SSH keys, except for the new key that you are currently using, from the list of authorized keys on the *Application* and *Monitor Servers*.

1.54.3 Rotate the *Submission Key*

The *Submission Private Key* is held on the airgapped *Secure Viewing Station*, and is not normally accessed by SecureDrop users anywhere but on the *Secure Viewing Station*. Therefore, we recommend rotating the *Submission Key* under the following circumstances:

- If the user's departure was not amicable
- If the user is still holding on to any *Secure Viewing Station* USB flash drive or backup
- If you have any other reason to believe the *Submission Private Key* or the entire *Secure Viewing Station* USB flash drive may have been copied or compromised.

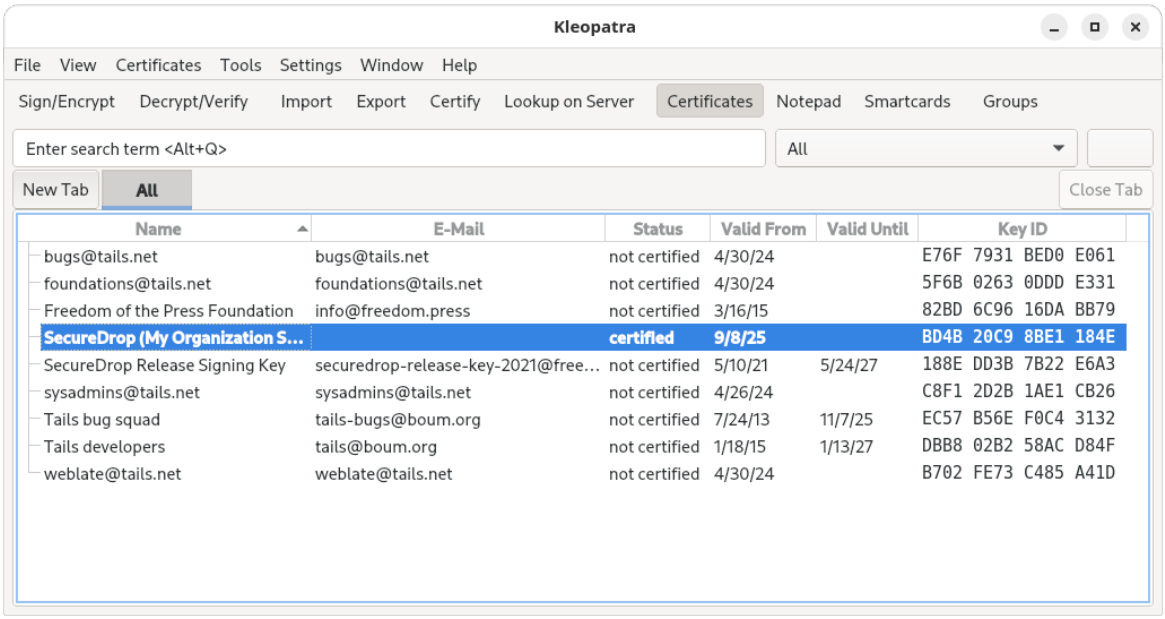
You should still keep the old key on the *Secure Viewing Station*, or else you will not be able to decrypt submissions that were sent to you while that key was in effect.

You will need:

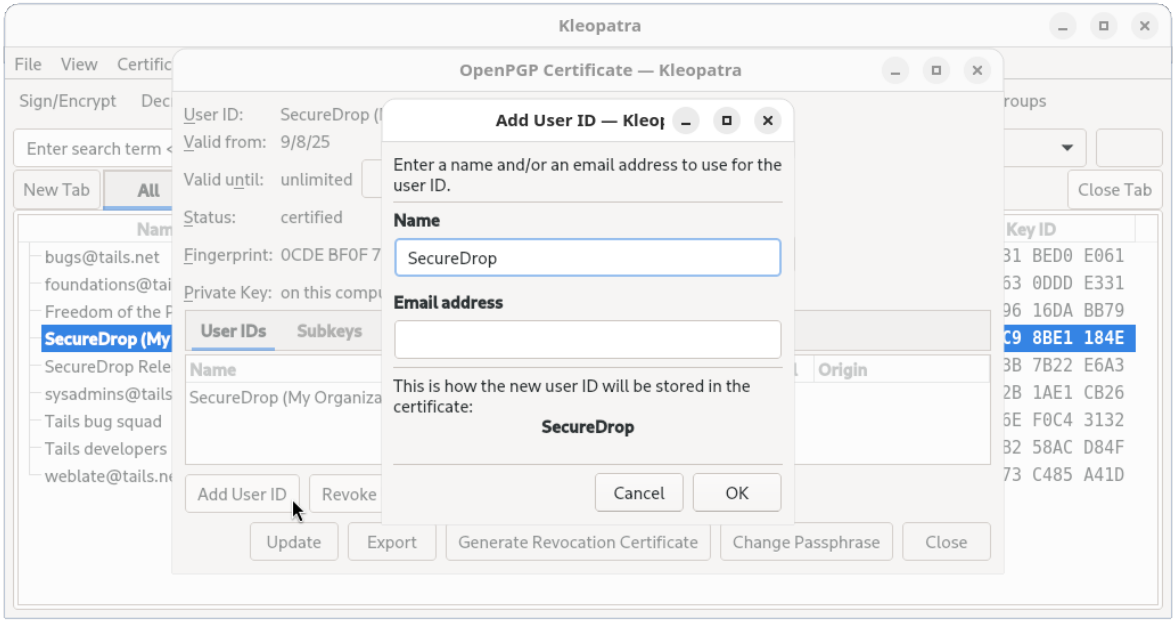
- A *Transfer Device* (LUKS-encrypted USB flash drive)

On the *Secure Viewing Station*

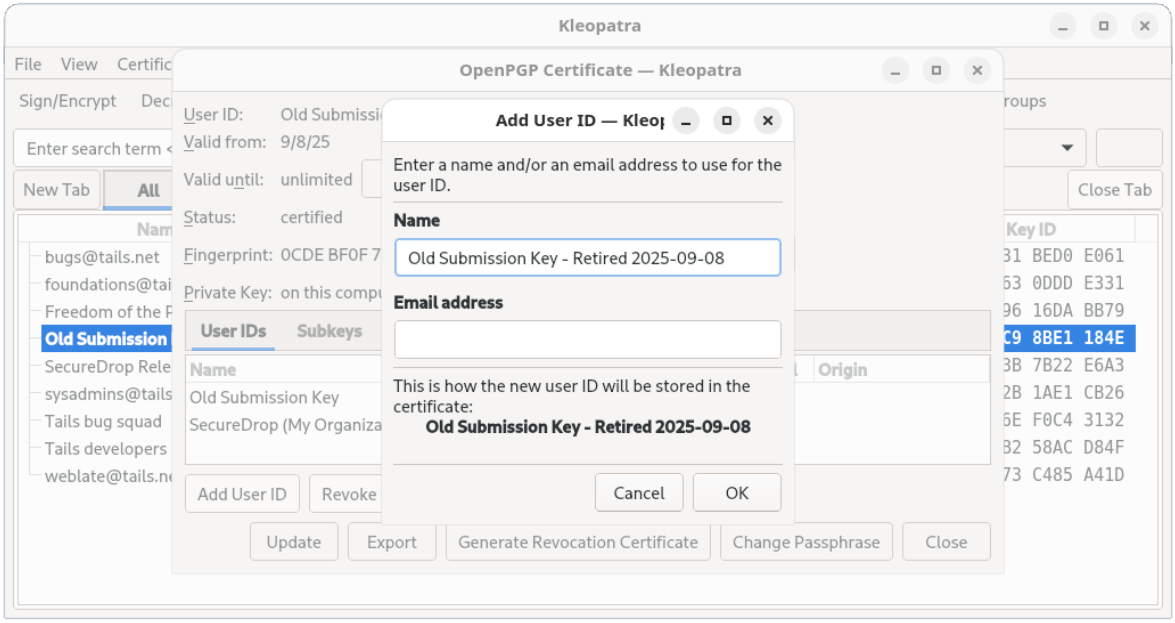
1. From the *Secure Viewing Station* Apps Menu, choose **Accessories ▶ Kleopatra**, and select the *Submission Key* from the list of available keys.



2. From the details view that appears, click the **Add User ID** button.



3. Set the name field to “Old SecureDrop Submission Key - Retired “, and add the date of retirement. Click **OK** to add this information to the key.



Note

This is a local-only change to stop you from mixing up the old and new keys

4. Return to the Terminal, then run:

```
gpg --list-keys
```

In the output, locate the "Old SecureDrop Submission Key". It should look similar to this:

```
pub   rsa4096/0x1CB396626CA370AB 2022-08-16 [SC]
      Key fingerprint = 6A7F 116B 3C22 4F36 7275 236A 1CB3 9662 6CA3 70AB
uid         [ultimate] Old SecureDrop Submission Key (Retired 2022-08-16)
uid         [ultimate] SecureDrop (SecureDrop Submission Key)
sub   rsa4096/0x228C92459E3D16DE 2022-08-16 [E]
```

Make note of the ID of the key, which is the portion of the key after the slash in the first line. In this example, the key ID would be: `0x1CB396626CA370AB`

5. Generate a revocation certificate, by running the command below (replacing `<KEY_ID>` with the ID you noted in the step above):

```
gpg --output revoke.asc --gen-revoke <KEY_ID>
```

This will launch an interactive prompt, where you can supply the following values:

```
Create a revocation certificate for this key? (y/N) y
Please select the reason for the revocation:
  0 = No reason specified
  1 = Key has been compromised
  2 = Key is superseded
  3 = Key is no longer used
  Q = Cancel
```

(continues on next page)

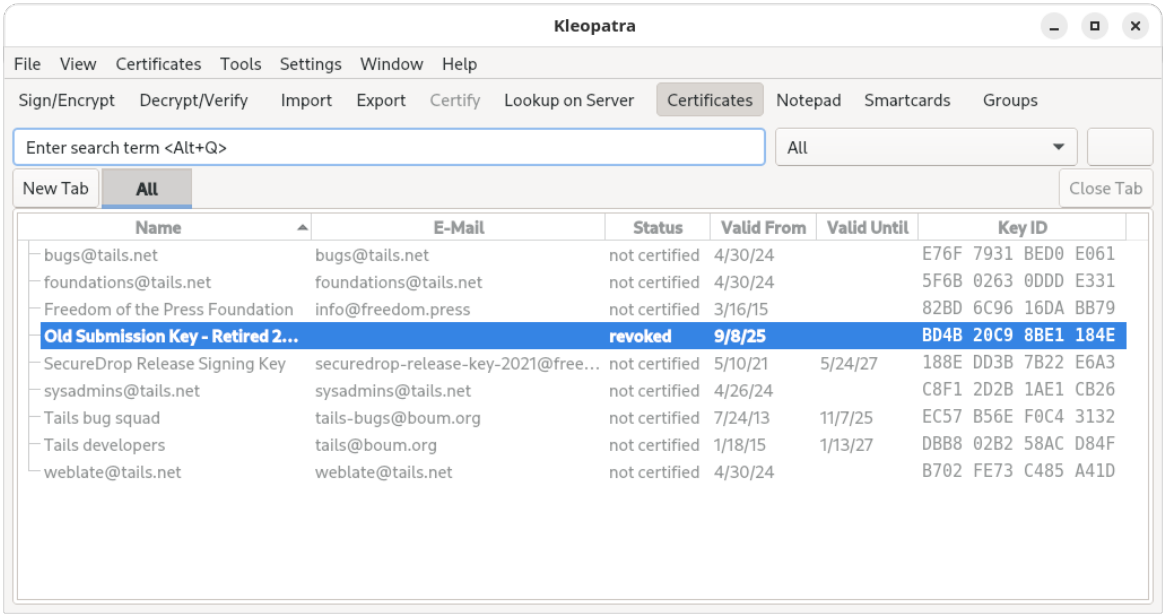
(continued from previous page)

```
(Probably you want to select 1 here)
Your decision? 2
Enter an optional description; end it with an empty line:
> <Just Press Enter>
Reason for revocation: Key is superseded
(No description given)
Is this okay? (y/N) y
ASCII armored output forced.
Revocation certificate created.
```

6. Import the revocation certificate:

```
gpg --import revoke.asc
```

7. Return to Kleopatra, and make sure the key is now marked as **Revoked**.



8. Now *follow the instructions* to create a new *Submission Key*. Copy the fingerprint and new *Submission Public Key* to your *Transfer Device*.

1.55 The `securedrop-admin` Utility

1.55.1 Using `securedrop-admin`

The `securedrop-admin` command-line utility is used to perform common server administration tasks, including:

- configuring and installing SecureDrop
- backing up and restoring the servers (see *Backing up and restoring servers*)
- retrieving server logs for troubleshooting (see *Investigating logs*)
- updating your SecureDrop servers' configuration post-install.

You can list all available `securedrop-admin` actions using the command `securedrop-admin --help`

Note

If your team has multiple admins, each with their own *Admin Workstation*, you must take steps to manually synchronize any configuration changes made via `securedrop-admin` with each other. See *Managing Configuration Updates with Multiple Admins*

1.55.2 Updating the server configuration

There are two primary reasons why you may want to update the system configuration:

- to change SecureDrop server configuration options. **Example:** You want to change the time of day at which the servers are automatically rebooted (default: 4:00 AM).
- to restore a valid configuration state on your servers. **Example:** Another admin has directly modified the iptables rules during troubleshooting, and you want to reinstate the correct rules.

In both cases, follow these steps:

1. Determine the current version of *securedrop-admin* you have installed on your *Admin Workstation* by running:

   ```
   apt-cache policy securedrop-admin
   ```

   If the version is the same as the version number displayed in the footer of your *Source Interface*, you are running the applicable version of the SecureDrop code on your workstation, and can proceed to the next step.

   If the versions differ, **it is not safe to proceed**. Follow the upgrade instructions associated with the release notes for the most recent release of SecureDrop. Apply all available updates, including for the Tails operating system.

1. Run `securedrop-admin sdconfig`. This will display the current configuration, one line at a time, and allow you to change it. At this point, any changes you make are only saved on this *Admin Workstation*, to the following file:

   `~/.config/securedrop-admin/site-specific`

2. Run `securedrop-admin install`. This will apply the configuration to your *Application* and *Monitor Server*, and enforce the canonical state of the server configuration.

Note

If you see an error running `securedrop-admin install`, and believe it may be an intermittent issue (for example, due to losing network connectivity to the servers), it is safe to run the `securedrop-admin install` command again. If you see the same issue consistently, then you will need to troubleshoot it.

If you see the error message "timeout (62s) waiting for privilege escalation prompt", try deleting the Ansible control path directory on your *Admin Workstation* (`rm -rf ~/.ansible/cp`) to reset the connection to the servers, then re-run the `securedrop-admin install`.

If you encounter other errors, we encourage you to submit a bug report, or to contact us at securedrop@freedom.press (GPG encrypted).

1.55.3 Updating localization for the *Source Interface* and the *Journalist Interface*

The *Source Interface* and *Journalist Interface* are translated in the following languages:

https://github.com/freedomofpress/securedrop/blob/develop/securedrop/i18n.rst

At any time during and after initial setup, you can choose from a list of supported languages to display using the codes shown in parentheses.

Note

With a *Source Interface* displayed in French (for example), *Sources* submitting documents are likely to expect a *Journalist* fluent in French to be available to read the documents and follow up in that language.

To add or remove locales from your instance, you'll need to *update your system configuration* as outlined above.

When you reach the prompt starting with "Space separated list of additional locales to support", you will see a list of languages currently supported. Refer to the list above to see which languages correspond to which language codes. For example:

```
Space separated list of additional locales to support (ru nl pt_BR fr_FR tr it_IT zh_
↪Hant sv hi ar en_US de_DE es_ES nb_NO): nl fr_FR es_ES
```

You'll need to list all languages you now want to support, adding or removing languages as needed. Locale changes will be applied after the next reboot.

1.55.4 Managing configuration updates with multiple admins

Organizations with multiple admins should set up a way to synchronize any changes one admin makes to the server configuration, as by default those changes are stored only on their individual *Admin Workstation*.

Configuration changes will be flagged by OSSEC and will generate alerts, but if other admins don't regularly review OSSEC alerts they may miss important changes, such as an update to the *Submission Public Key*. If they subsequently run `securedrop-admin install` from their *Admin Workstation*, they will revert the server configuration to an older version.

The simplest approach to keeping workstations in sync is to inform other admins of changes as you make them, for example via a secure Signal group chat. Any such communications should happen over a platform that provides E2EE, as you may need to share sensitive information.

Configuration information is stored on the *Admin Workstation* under `~/.config/securedrop-admin`:

- `~/.config/securedrop-admin/site-specific` contains settings written by `securedrop-admin sdconfig` - if it is changed other admins should be notified.
- The *Submission Public Key* and *OSSEC Alert Public Key* should be present under `~/.config/securedrop-admin`. If these keys are rotated, the public keys should be updated on other *Admin Workstations*.
- *Onion Service* information is stored in several files:

  ```
  ~/.config/securedrop-admin/app-ssh.auth_private
  ~/.config/securedrop-admin/mon-ssh.auth_private
  ~/.config/securedrop-admin/app-journalist.auth_private
  ~/.config/securedrop-admin/app-sourcev3-ths
  ~/.config/securedrop-admin/tor_v3_keys.json
  ```

  If *Onion Service* addresses are changed, the files listed above should be shared securely with other administrators - preferably in person using an encrypted USB flash drive, as they can be used to access the servers directly via SSH over Tor.

1.56 Upgrade guide

1.57 Investigating logs

When troubleshooting issues with your SecureDrop instance, be sure to examine all relevant log files on both servers. To work with logs, it is helpful to be familiar with commands like `less`, `tail` and `grep`; to inspect older, archived logs (names end with `.gz`) you can use commands like `zless` and `zgrep` to avoid manually decompressing each file.

Note

You can use the `securedrop-admin` tool to extract logs to send to Freedom of the Press Foundation for analysis. Run the following command:

```
securedrop-admin logs
```

This command will produce encrypted tarballs containing logs from each server. See the command output for more information.

1.57.1 Logs to examine on both servers

- `/var/log/kern.log`: Use this file to investigate kernel-related issues, including warnings or errors specific to AppArmor or grsecurity (a set of patches applied to the kernels for additional security hardening)
- `/var/log/syslog`: Use this file to investigate most other system issues, including iptables configuration problems or Tor network issues. Use search patterns, e.g., search for "app Tor" to find log entries specific to Tor.

1.57.2 *Application Server* logs

See the directory `/var/log/apache2/*` for web server access and error logs. In production systems, logging is only enabled for the *Journalist Interface* to the files `journalist-access.log` and `journalist-error.log`, and the logs do not contain IP address information.

When investigating an application error on the *Source Interface* (e.g., if you see an "Internal Server Error" when submitting a document), it can make sense to temporarily enable error logging. To do so:

1. Log into your *Application Server* from your *Admin Workstation* via `ssh app`
2. Edit the file `/etc/apache2/sites-enabled/source.conf` (requires `sudo`)
3. Comment out the old `ErrorLog` and `LogLevel` directives, e.g., like so:

```
# Enabling logging for error investigation, 2020-04-18, ~admin
#
# ErrorLog /dev/null
# LogLevel critical
```

4. Add the desired new logging configuration in the same location (inside the <`VirtualHost`> block), e.g.:

```
ErrorLog /var/log/apache2/source-error.log
LogLevel debug
```

5. Save the file and reload the configuration with `sudo systemctl reload apache2`
6. Visit the *Source Interface* and reproduce the error
7. Inspect the log file `/var/log/apache2/source-error.log` for any details
8. Remember to set the configuration back to the default values once your investigation is complete.

Note that the `debug` logging level is highly verbose; if you want to adjust it, see the Apache documentation for more information about the different logging levels.

If you encounter an application error, and you have not modified the application code, please be sure to file an issue or contact us via securedrop@freedom.press (GPG encrypted).

1.57.3 *Monitor Server* logs

- `/var/ossec/logs/ossec.log`: Examine this file to investigate problems with OSSEC itself not functioning as expected (e.g., you are not seeing alerts when you would expect them to).
- `/var/ossec/logs/alerts/alerts.log`: This file contains the most recent alerts generated by OSSEC. If you have correctly configured OSSEC emails, the text of these alerts should correspond to the text of the emails.
- `/var/log/mail.log` and `/var/log/procmail.log`: These files contain information about email delivery and email processing (for encrypting the alerts). Investigate these files if you believe OSSEC is correctly configured, but you are not receiving emails.

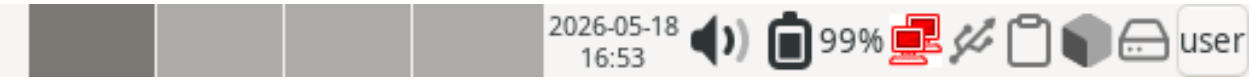
1.58 Troubleshooting connection problems

Before troubleshooting connection problems, we recommend reading about the *networking architecture* of SecureDrop Workstation. If you are in a hurry, this guide offers quick diagnostic and remedial steps.

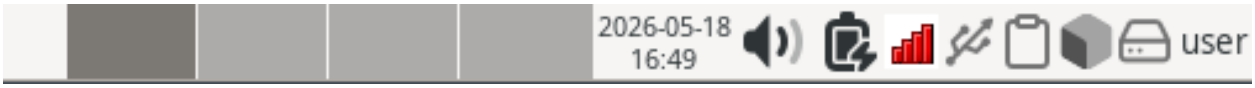
1.58.1 Step 1: Verify you are connected to the Internet

You can use both wireless and wired networks in Qubes. You can manage network access through the network manager, which you can find in the area populated with icons in the top right corner of your Qubes desktop, known as the *system tray*.

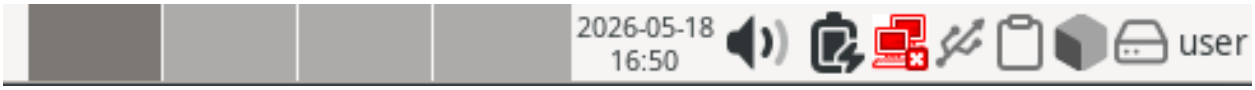
The network manager is the red icon, which looks like this for a wired connection (ordering of icons may vary):



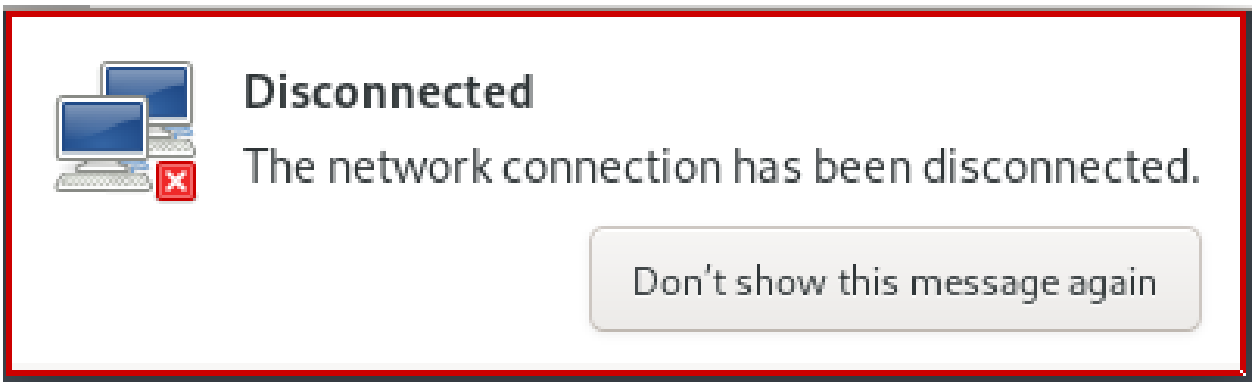
It looks like this for a wireless connection:



It looks like this when you are not connected to the Internet at all:



When a network connection is lost, Qubes will display an alert like the following:

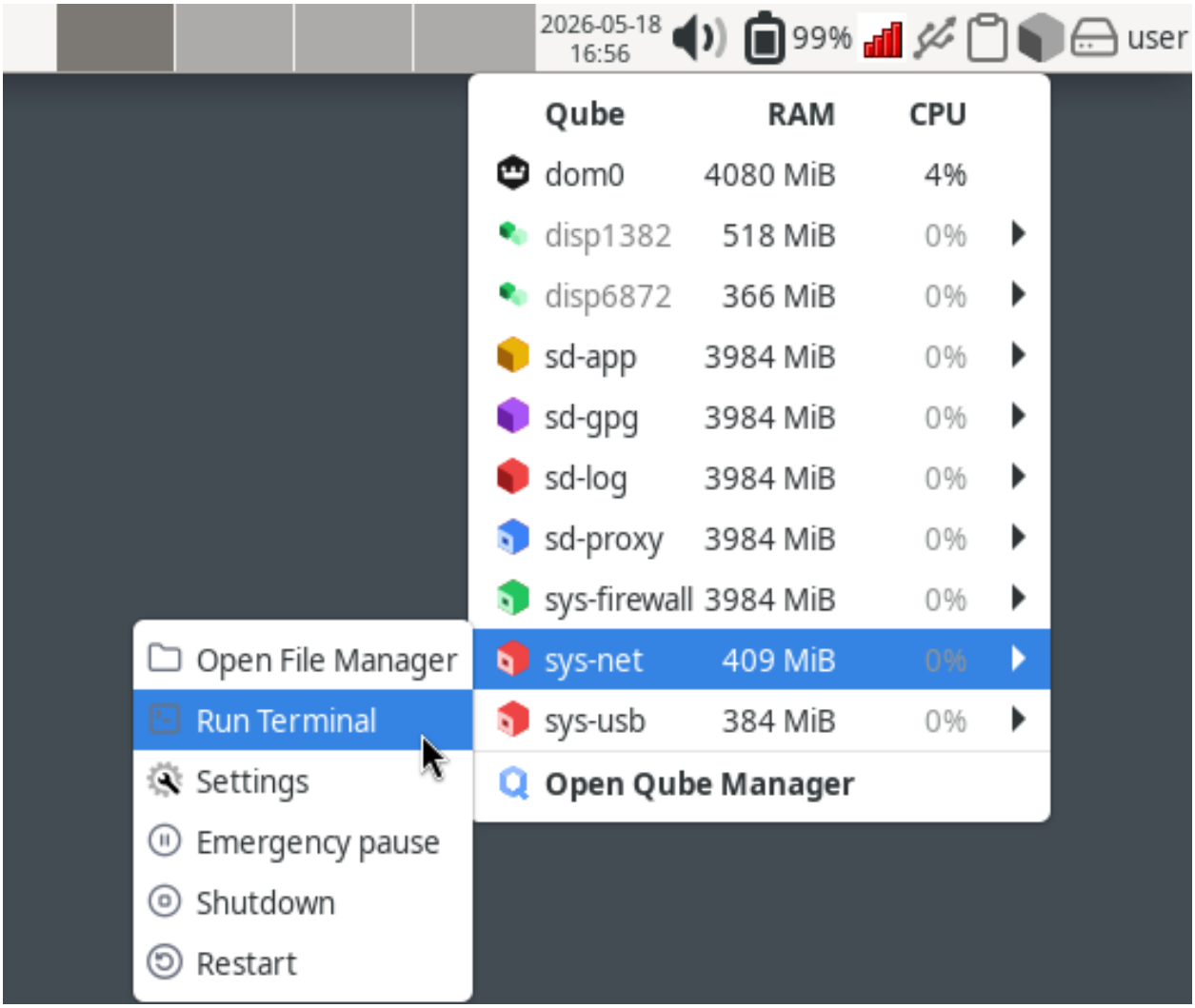


Common causes for lost connections include fully or partly unplugged network cables, lost power to networking equipment, and ISP service outages. When you see a lost connection notification, it is most likely due to one of these causes.

Important

Not all VMs in Qubes OS have Internet access. For example, opening a terminal via ▶ ▶ **Other Tools** ▶ **Xfce Terminal** opens a `dom0` terminal without Internet access. See our *networking architecture* overview for additional background.

If the network manager shows that you are connected to the Internet, you can verify whether your connection is working by opening a terminal in `sys-net`:



1. Click the Qubes Domains menu in the in the system tray (top right area).
2. A list of running VMs should appear. Select `sys-net` from the list, and click **Run Terminal**.
3. In the terminal window, type the command `ping -c 5 google.com`.

You should see a sequence of lines starting with `64 bytes from` and ending with the number of milliseconds it took to complete the request. If you do not see similar output, your network access may be misconfigured, or the Internet may be wholly or partially unreachable. If using `8.8.8.8` instead of `google.com` works, it may suggest a problem at the DNS level in your network configuration.

If you have verified that you are able to connect to the Internet using `sys-net`, but you are experiencing other connectivity issues, move on to the next step.

1.58.2 Step 2: Troubleshooting login issues

Issues logging in may not be network-related. If you are experiencing connectivity issues before or after logging in, you can skip ahead to the next section.

Make sure that your username, passphrase, and two-factor code are correct.

Important

After a failed login, wait for a new two-factor code from your app before trying again.

You can reveal the passphrase by clicking the "eye" icon next to it in the login dialog (ensure you are in a fully private setting before doing so). Check for extra characters and end, or subtle differences like capitalization. Note that the spaces between words in SecureDrop passphrases are part of the passphrase.

If you use the two-factor app on your phone for other websites and services, make sure that you have selected the correct user account. It should be labeled **SecureDrop**.

If you have access to a Tails-based *Journalist Workstation*, verify whether you can access SecureDrop from Tails.

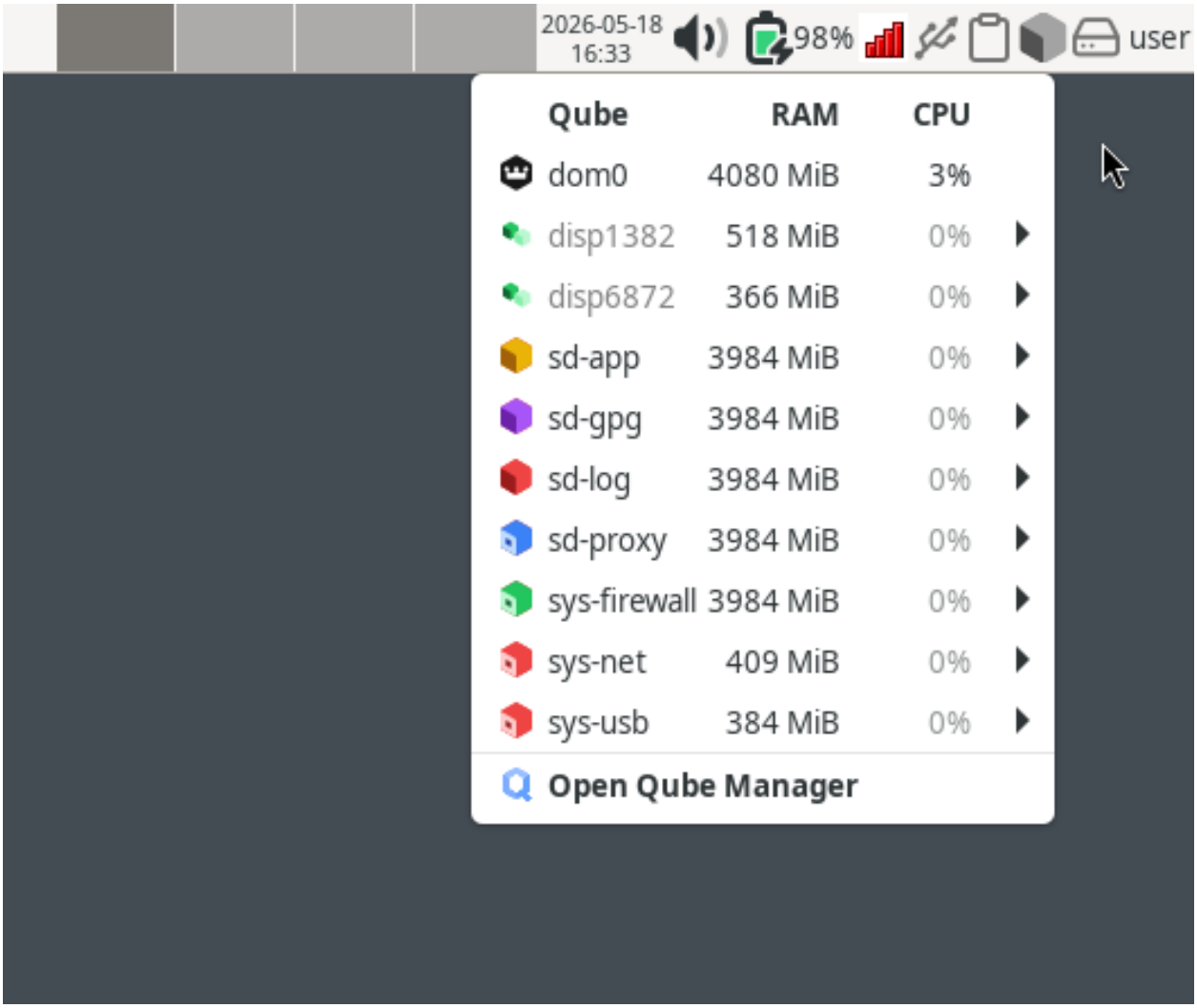
If you are certain that your credentials are correct but you are unable to log in, proceed to the next step.

1.58.3 Step 3: Verify that all required VMs are running

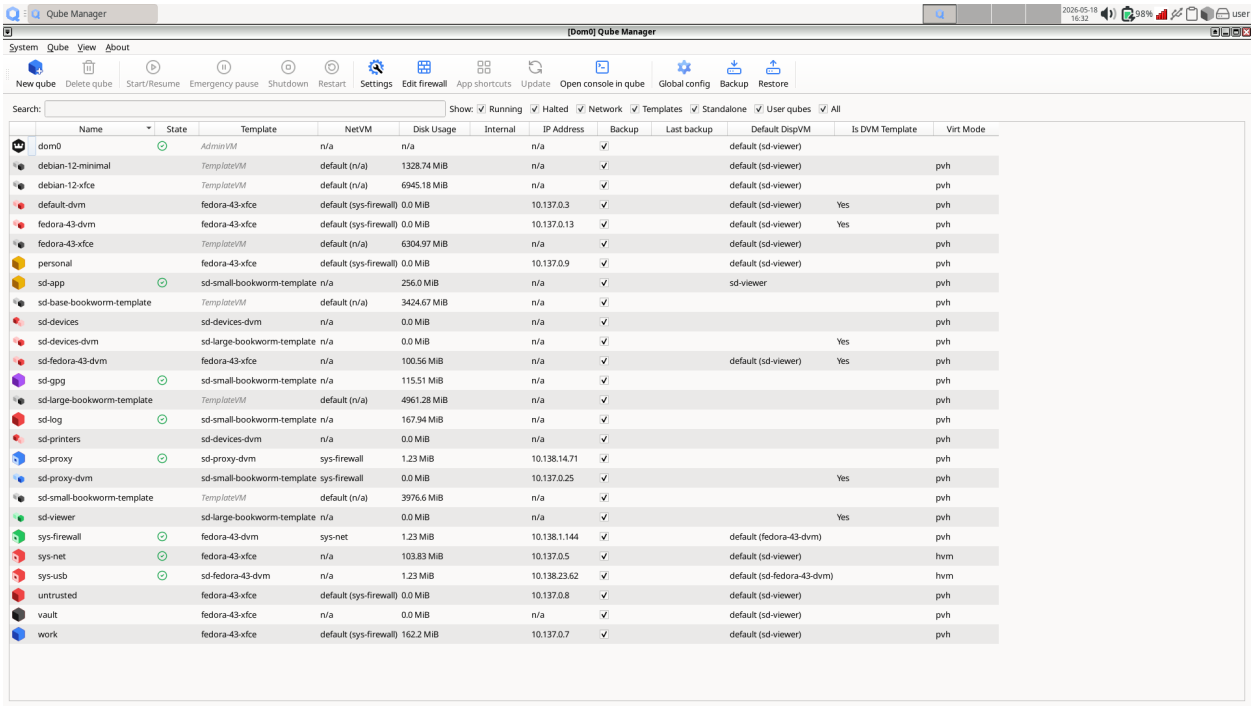
The following VMs must be running for all actions requiring network connectivity to work (e.g., logging in, checking for messages, downloading documents, replying to sources, starring sources, deleting sources):

- `sd-app`
- `sd-gpg`
- `sd-log`
- `sd-proxy`
- `sys-firewall`
- `sys-net`

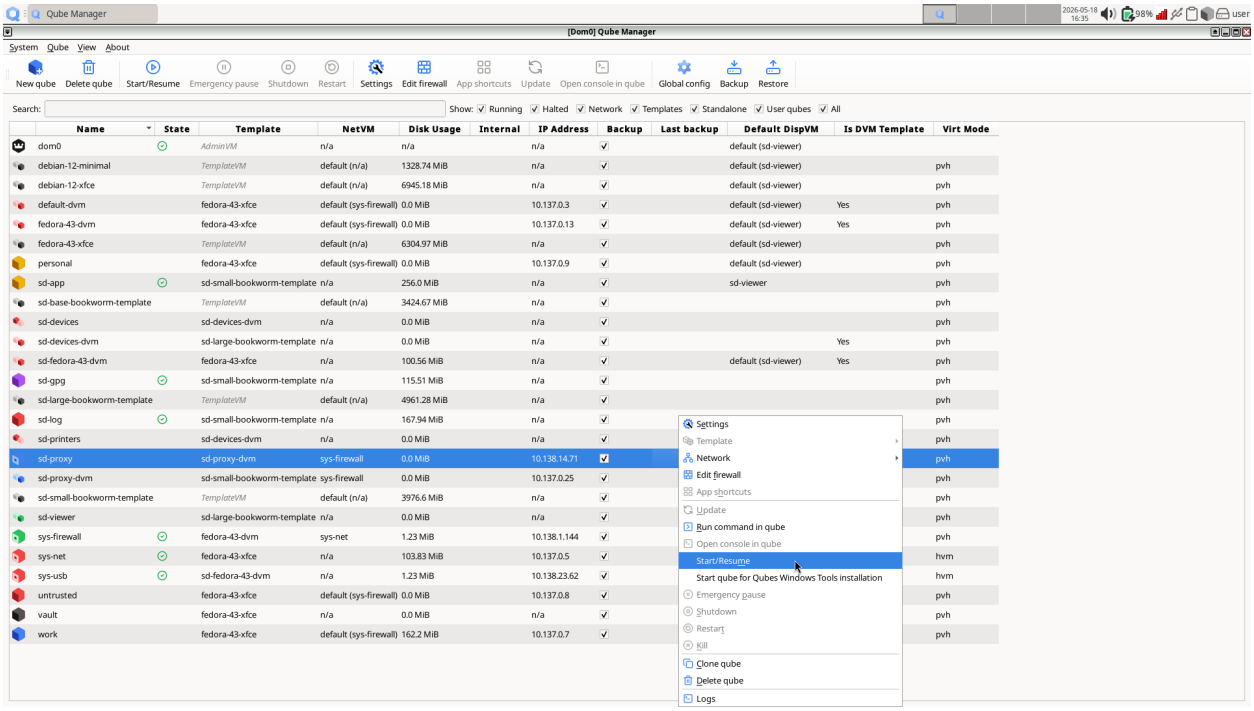
You can verify whether a VM is running or not by clicking the icon in the system tray (top right). Only VMs that are currently running will appear in the list:



If a required VM is not running, you can launch it from the Qube Manager. Open the Qube Manager by clicking **Open Qube Manager** in the menu above. A window like the following should appear:



To start a VM, select it from the list, right-click it, and click **Start/Resume Qube**. Alternatively, you can click the "Play" button in the toolbar.



In ordinary use, VMs required by SecureDrop should be started on boot or when they are needed. If you repeatedly experience problems with a necessary VM not running, or if an error message is displayed when attempting to start the VM, please contact us for assistance.

If all required VMs are running, proceed to the next step.

1.58.4 Step 4: Verify that required VMs have connectivity

In step 1, you have already verified that you can connect to the Internet using `sys-net`. Now, test whether `sys-firewall` and `sd-proxy` are working.

First, open a terminal in `sys-firewall` and run the `ping google.com` command. You should see similar output as in `sys-net` before.

Now, open a terminal in `sd-proxy` and run the following command:

```
curl -s --proxy socks5h://localhost:9150 https://check.torproject.org | grep
Congratulations
```

This command contacts a service intended for web browsers to verify whether your Tor connection is working.

You should see the text "Congratulations. This browser is configured to use Tor." or a similar message on the terminal.

If the output does not include the text "Congratulations", proceed to the next steps.

1.58.5 Step 5: Restart `sd-proxy`

Restart `sd-proxy` to attempt to restore connectivity:

1. Exit SecureDrop Inbox if it is running.
2. Click the Qubes Application menu  icon in the system tray (top left).
3. Click **Run Qube Manager**
4. Right-click `sd-proxy` in the list of VMs. Click **Shutdown qube**.
5. Right-click `sd-proxy` in the list of VMs. Click **Start/Resume qube**.

If this does not resolve the issue, proceed to the next step.

1.58.6 Step 7: Restart `sys-net` and `sys-firewall`

Note

You will temporarily lose all Internet connectivity in Qubes OS during this step.

Using the same procedure as in the previous step, shut down `sd-proxy`. Attempt to shut down `sys-firewall`. You may see an error message telling you that other VMs still require access to `sys-firewall`. Save your work in those VMs, shut them down, and attempt to shut down `sys-firewall` again.

Finally, shut down `sys-net`. The network manager icon should disappear.

Now, start `sys-proxy`, which will bring up `sys-net` and `sys-firewall` at the same time.

If this does not resolve the issue, please contact us for assistance.

1.58.7 Examining logs

You may wish to examine system logs on your own, or with our guidance. You can examine consolidated syslogs from all SecureDrop-related VMs in the `sd-log` VM. They can be found in the default user's `~/QubesIncomingLogs` directory.

In addition, you may want to examine `/var/log/syslog` in `sys-net` and `sys-firewall`.

1.58.8 I can't SSH into my servers over Tor. What do I do?

At any point after the successful installation of SecureDrop, if you cannot SSH into your servers, you should first perform the following troubleshooting steps:

1. **Ensure that you are connected to Tor.**
2. **Ensure your servers are online.** Visit the *Admin Interface* to check your *Application Server* is online, and you can trigger a *test OSSEC alert* to verify your *Monitor Server* is online.
3. **Ensure that SSH aliases and *Onion Service* authentication are configured:**
   - First, ensure that the correct configuration files are present in `~/.config/securedrop-admin`:
     - `app-ssh.auth_private`
     - `mon-ssh.auth_private`
     - `app-journalist.auth_private`
     - `app-sourcev3-ths`
     - `tor_v3_keys.json`
   - Then, run `securedrop-admin localconfig`. This will ensure your local Tails environment is configured properly.
4. **Confirm that your SSH key is available**: During the install, you configured SSH public key authentication using `ssh-copy-id`. Ensure this key is available using `ssh-add -L`. If you see the output "This agent has no identities." then you need to add the key via `ssh-add` prior to SSHing into the servers.

1.59 Backing up and restoring servers

Maintaining regular backups helps guard against data loss and hardware failure. Having a recent backup will allow you to redeploy SecureDrop without changing onion addresses, recreating journalist accounts, or losing previous submissions from *Sources*.

Note

Only the *Application Server* is backed up and restored, including historical submissions and both *Source Interface* and *Journalist Interface* URLs. The *Monitor Server* needs to be configured from scratch in the event of a hardware migration.

1.59.1 Minimizing disk use

Since the backup and restore operations both involve transferring *all* of your SecureDrop's stored submissions over Tor, the process can take a long time.

Encouraging *Journalists* to regularly delete older, unneeded submissions from the *Journalist Interface* will save time and improve reliability when doing backups.

Tip

Although it varies, the average throughput of an *Onion Service* is about 3 Mbps, or roughly 90 minutes for 2GB. Plan your backup and restore accordingly.

On the *Application Server*, open a Terminal via **Apps ► System Tools ► Console** on the *Admin Workstation* and run

```
ssh app sudo du -sh /var/lib/securedrop/store
```

Compare the output of this command (which approximates the size of a backup archive) to the amount of free space on your Tails persistent volume via Tails' **Disks** utility to ensure you have sufficient space to perform a backup.

If you find you cannot perform a backup or restore due to this constraint, and have already deleted old submissions from the *Journalist Interface*, contact us through the SecureDrop Support Portal.

Note

Submissions are deleted asynchronously and one at a time, so if you delete a lot of submissions through the *Journalist Interface*, it may take a while for all of the submissions to actually be deleted. SecureDrop uses `shred` to securely erase files, which takes significantly more time than normal file deletion. You can monitor the progress of queued deletion jobs by logging in to the *Application Server* over SSH and running:

```
sudo journalctl -u securedrop_rqworker
```

1.59.2 Backing up

Check connectivity

Open a Terminal via **Apps ▶ System Tools ▶ Console** on your *Admin Workstation* and verify it is able to run Ansible and connect to the SecureDrop servers.

```
ssh app uptime
```

If this command fails, see *Troubleshooting*.

Create the backup

When you are ready to begin the backup, run

```
securedrop-admin backup
```

The backup command will display updates on its progress as the backup is created. Run time will vary depending on connectivity and the number of submissions saved on the *Application Server*.

When the backup action is complete, the backup will be stored as a compressed archive in `~/.config/securedrop-admin`. The filename will begin `sd-backup`, followed by a timestamp of when the backup was initiated, and ending with `.tar.gz`. You can find the full path to the backup archive in the output of the backup command.

Warning

The backup file contains sensitive information! It should only be stored on the *Admin Workstation*, or on a dedicated encrypted backup USB flash drive.

Note

When dealing with larger backups, the `securedrop-admin backup` command may fail with a `MemoryError` at this stage of the operation: "Fetch the backup tarball back to the Admin Workstation".

If this happens, a backup was successfully generated, but it is still on the server. Run this command from your `~/.config/securedrop-admin` directory to copy the backup your *Admin Workstation*:

```
rsync -av --progress --partial app:$(ssh app ls -1rt /tmp/sd-backup* | tail -1) ~/.
↪config/securedrop-admin/
```

If the transfer fails or is interrupted, you can simply run this command again to resume it.

Note that this method will only work if you have first run the `securedrop-admin backup` command, and the backup has successfully progressed at least until the "Fetch the backup tarball" stage.

1.59.3 Restoring from a backup

Prerequisites

To perform a restore, boot into the *Admin Workstation* and ensure that your `.tar.gz` backup archive has been copied to `~/.config/securedrop-admin`. (If you are using the same *Admin Workstation* as you did when you took the backup, the archive will already be in place).

If you are restoring data onto an existing instance (for example, for data recovery purposes), see *Restoring a Backup on an Existing Instance*.

If you are reinstalling SecureDrop and then restoring from a backup (for example, for hardware migration, operating system upgrade, or disaster recovery purposes), see *Migrating Using a Backup*.

For other data recovery scenarios, see *Additional Information* or contact Support.

Restoring a backup on an existing instance

To restore an existing instance to a previous state, run the command:

```
securedrop-admin restore sd-backup-2020-07-22--01-06-25.tar.gz
```

Make sure to replace `sd-backup-2020-07-22--01-06-25.tar.gz` with the filename for your backup archive.

This command attempts to restore submissions, source and journalist accounts, and configuration details for the *Onion Services* used by the web interfaces and SSH (if configured).

1.59.4 Migrating using a backup

Moving a SecureDrop instance to new hardware involves:

- Backing up the old instance and preserving configuration and credentials from the *Admin Workstation*;
- Installing SecureDrop on new hardware;
- Restoring the backup to the new instance and repairing credentials.

Note

If you need to restore from a backup from an instance configured to use SSH-over-LAN onto an SSH-over-Tor instance, you must either first update the target instance to use SSH-over-LAN or perform a data-only backup. See *Data-only Restores* for more information.

Note

The instructions below assume that you are using the same *Admin Workstation* that was used to manage your old instance. If you are using a new *Admin Workstation* you will need to first install the securedrop-admin package and

prerequisites on it. Then you may copy the config directory `~/.config/securedrop-admin` and backup archive from the old *Admin Workstation* to the new workstation (using an encrypted *Transfer Device*), and proceed with the instructions below.

1. If you have not already done so, *back up the existing installation*. The instructions below assume that the backup has been created and renamed `sd-backup-old.tar.gz`.
2. Move the existing *Admin Workstation* SSH configuration out of the way via the Terminal via **Apps ▶ System Tools ▶ Console**, using the commands:

```
ssh-add -D
find ~/.ssh/ -type f -exec mv {} {}.bak \;
```

Note

You will be generating fresh SSH credentials for the servers, and any other *Admin Workstation* USB flash drives will have to be *provisioned with updated credentials*.

3. Ensure your *Admin Workstation* is connected to a LAN port on your network firewall, and *configure the Admin Workstation's IP address*.
4. Install Ubuntu 24.04 on the *Application* and *Monitor Servers*, following the *server setup instructions* to install with the correct settings, test connectivity, and set up SSH keys to allow for *Admin Workstation* access.

Note

You may need to wait approximately 10-15 minutes after installing Ubuntu 24.04 for the servers to become reachable via SSH.

5. Reinstall SecureDrop on the servers, following the *installation instructions*. During the configuration stage (`securedrop-admin sdconfig`), the values will be prepopulated based on the old instance's configuration, which is still stored in `~/.config/securedrop-admin`. Press **Enter** to accept each value.

   Proceed through the installation by running `securedrop-admin install` then `securedrop-admin localconfig`. If SSH-over-Tor is configured, run `ssh app uptime` and `ssh mon uptime` in the Terminal to verify SSH connectivity.
6. Restore from the old instance's backup (e.g. `sd-backup-old.tar.gz`) using the Terminal command:

```
securedrop-admin restore sd-backup-old.tar.gz
```

   The restore task will proceed for some time.
7. Synchronize the server and *Admin Workstation's* web interface config and authentication keys using the Terminal commands:

```
securedrop-admin install
securedrop-admin localconfig
```

8. *Test the new instance* to verify that the web interfaces are available and the servers can be reached via SSH.
9. If you have migrated to new hardware, ensure your old servers have been decommissioned and/or destroyed by following the relevant sections of *our decommissioning documentation*.

Repair additional *Admin Workstations*

If you have additional *Admin Workstation* USB flash drives, they will no longer have valid SSH credentials and will need to be repaired. In these steps, the "primary *Admin Workstation*" is the one which you used to complete the above migration process.

1. Prepare a fresh *LUKS-encrypted USB flash drive*. You may record the passphrase in your primary *Admin Workstation* KeePassXC password manager.
2. Copy the following files from your primary *Admin Workstation* onto the LUKS-encrypted USB flash drive:
   - `~/.config/securedrop-admin/tor_v3_keys.json`
   - `~/.config/securedrop-admin/mon-ssh.auth_private`
   - `~/.ssh/id_rsa.pub`
   - `~/.ssh/id_rsa`

   Note

   Alternatively, if you wish to use different SSH credentials for each *Admin Workstation*, you may do so. In this case, copy only the first two files above to your additional *Admin Workstations*.

   Generate per-machine SSH keys and use a clean LUKS-encrypted USB flash drive to transfer the public portions of those keys to your primary *Admin Workstation*, where you will then add them to the servers' `authorized_keys` files, as described *here*. You may also contact Support for assistance.
3. Boot into each additional *Admin Workstation*. Set an administration password and unlock the persistent volume on the Tails welcome screen. Once logged in, attach the LUKS-encrypted USB flash drive and unlock it.
4. Ensure that this *Admin Workstation* is using an up-to-date version of Tails and is running the latest SecureDrop application code, 2.14.0.
5. As you did with the primary *Admin Workstation*, archive the existing SSH configuration:

   ```
   ssh-add -D
   find ~/.ssh/ -type f -exec mv {} {}.bak \;
   ```
6. From the LUKS-encrypted USB, copy `~/.ssh/id_rsa` and `~/.ssh/id_rsa.pub` to the `~/.ssh/` directory.
7. From the LUKS-encrypted USB, copy `tor_v3_keys.json` and `mon-ssh.auth_private` to the `~/.config/securedrop-admin` directory.
8. In the Terminal, type the following commands:

   ```
   securedrop-admin localconfig
   ```
9. Test connectivity to each server by running `ssh app uptime` and `ssh mon uptime`.
10. Once all *Admin Workstations* have been updated, securely wipe the files on the LUKS-encrypted USB flash drive, by right-clicking them in the file manager and selecting **Wipe**. Then, reformat the device using the **Disks** utility.

1.59.5 Additional information

Data-only restores

The `restore` command normally restores both the data and the Tor configuration of an instance, including the onion addresses for your instance.

You may, however, restore data, such as submissions and journalist and source accounts, without altering an instance's Tor configuration, with the following command:

```
securedrop-admin restore --preserve-tor-config sd-backup-2020-07-22--01-06-25.tar.gz
```

If you require any assistance with migration or data recovery, please contact Support.

1.60 Rebuilding an *Admin Workstation*

In cases where an *Admin Workstation* USB flash drive has been lost or destroyed, and no backup exists, it is possible to rebuild one. In order to do so, you'll need

- physical access to the SecureDrop servers
- 2 USB flash drives:
  - Tails Template drive
  - 1 replacement *Admin Workstation* USB flash drive (USB3 and 16GB or better recommended)

The process requires experience with the Linux command line and Tails, and can take up to 3 hours. If a backup of the SecureDrop *Application Server* is available, *reinstalling the instance and restoring the backup* may be simpler. An outline of the steps involved in rebuilding an *Admin Workstation* is as follows:

1. Prepare the USB flash drives.
2. (Optional) Boot the *Application* and *Monitor Server* in single user mode and reset the shell admin account password.
3. Set up SSH access for the new *Admin Workstation*.
4. Retrieve SecureDrop configuration settings from the *Application* and *Monitor Server*.
5. Back up and configure the *Application Server*.
6. Run `securedrop-admin install` and `securedrop-admin localconfig` from the new *Admin Workstation*.
7. Configure SSH-over-TOR.
8. Complete post-rebuild tasks.

Important

The rebuild process involves temporarily removing `iptables` rules on the *Application* and *Monitor Servers*, weakening their security. Because of this, it's important to complete the rebuild process promptly, to avoid leaving the servers in an insecure state.

1.60.1 Step 1: Prepare the USB flash drives

First, create a new Tails drive and set up a persistent volume with a strong passphrase.

Once persistence has been set up, start up the *Admin Workstation* with persistence enabled, install the SecureDrop Inbox code, and set up the KeePassXC database.

The *Admin Workstation* uses SSH with key authentication to connect to the servers, so you'll need to create a new SSH keypair for your SecureDrop instance. To do so, open a terminal by navigating to **Apps ▶ System Tools ▶ Console**, and run the following command:

```
ssh-keygen -t rsa -b 4096
```

When prompted to `Enter file in which to save the key`, Press **Enter** to use the default location. When prompted for a passphrase, it's safe to leave it blank.

1.60.2 Step 2: (Optional) Boot the servers in single-user mode

If you do not have the original password for the shell admin account on the *Application* and *Monitor Servers*, you'll need to reset the password on each server by booting in single user mode. In order to do so, you'll need physical access to the server, a keyboard, and a monitor.

First, connect a monitor and keyboard to the *Monitor Server*. Then reboot the server. Enter the GRUB menu (instructions vary by hardware), ensure the **Ubuntu** entry is highlighted, and press **e** to edit boot options.

In the boot options for Ubuntu, find the line that starts with `linux` and ends with `noefi ipv6.disable=1 quiet`. Add `single` after `quiet`, separated by a space, and press **F10** to boot in single user mode.

Reset the SecureDrop admin user's password

Once the root prompt appears, you'll need to reset the password for the SecureDrop admin user. By default this user is named *sdadmin* and has UID 1000. However it may have been named differently during the installation of your instance. You can use the command `getent passwd 1000` to check the username corresponding to UID 1000. Once you have the correct username, reset its password using the *passwd* command, for example:

```
passwd sdadmin
```

Important

Make sure to select a strong password, and record it in the *Admin Workstation's* KeePassXC database.

Finally, reboot the *Monitor Server* and verify that you can log in at the console using the new password.

Repeat the process for the *Application Server*. Use the same username and password as for the *Monitor Server* - this is required in order for the `securedrop-admin install` command to work correctly.

1.60.3 Step 3: Set up *Admin Workstation* access

Next, you'll configure the servers to allow temporary SSH access from the new *Admin Workstation*.

First, start the new *Admin Workstation* with persistence enabled and an administration password set.

Next, connect the new *Admin Workstation* to the *Hardware Firewall* via the appropriate Ethernet port, and set up its static IP address. For more information on how to do so, see *this section in the firewall setup documentation*. If you do not know the correct static IP address for the *Admin Workstation*, and you are using a recommended pfSense-based *Hardware Firewall*, you can retrieve the address by logging into its admin interface and checking the settings under **Firewall ► Aliases**.

Note

If you do not have login credentials for your pfSense firewall, check its user manual for instructions on resetting the administration password.

Next, determine whether your instance was set up to allow administrative access via SSH over Tor, or via SSH over LAN. If you don't know which option was originally chosen, you can check as follows:

1. Log in to the *Application Server* via the console using the adminstration username and password.

2. Check to see if an SSH hidden proxy service exists, using the command `sudo cat /var/lib/tor/services/sshv3/hostname`. If this file exists and includes an onion address, your instance is set up to use SSH over Tor and you should configure temporary SSH access using *these instructions*. If not, your instance is set up to use SSH over LAN, and you should follow *these instructions instead*.

Configuring access for an SSH-over-Tor instance

Direct SSH access is disabled when the SSH-over-Tor option is selected during installation. To temporarily re-enable it, you'll need to update `iptables` rules and change the sshd daemon's configuration.

First, log on to the *Application Server* via the console, and run the following commands, substituting the *Admin Workstation's* static IP for `<admin_static_ip>`:

```
sudo iptables -I INPUT -p tcp --dport 22 -s <admin_static_ip> \
  -m state --state NEW,ESTABLISHED -j ACCEPT
sudo iptables -I OUTPUT -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

Next, edit the file `/etc/ssh/sshd_config`, changing the line:

```
ListenAddress 127.0.0.1:22
```

to:

```
ListenAddress 0.0.0.0:22
```

and deleting the line:

```
PasswordAuthentication no
```

Restart `sshd` using the command `sudo systemctl restart ssh`.

Then, use the command `ip a` to note the local IP address of the default Ethernet interface. You'll need it in the next step.

Repeat the process above for the *Monitor Server*, making sure to note its local IP address as well.

Once the *Monitor Server* has been configured, proceed to *enable access from the new Admin Workstation*.

Configuring access for an SSH-over-LAN instance

First, log on to the *Application Server* via the console and edit the file `/etc/ssh/sshd_config`, deleting the line:

```
PasswordAuthentication no
```

Restart `sshd` using the command `sudo systemctl restart ssh`.

Then, use the command `ip a` to note the local IP address for the default Ethernet interface. You'll need it in the next step.

Repeat the process above for the *Monitor Server*, making sure to note its local IP address as well.

Enabling access from the new *Admin Workstation*

From the *Admin Workstation*, open a terminal and copy the *Admin Workstation*'s SSH public key to the servers, substituting the values for the server administration username and server IP addresses in the commands below and entering the admin account's password when prompted:

```
ssh-copy-id <admin-username>@<application-server-ip>
ssh-copy-id <admin-username>@<monitor-server-ip>
```

Next, create a file `~/.ssh/config` with contents as below, again substituting the appropriate values for your servers:

```
Host app
  User <admin-username>
  Hostname <application-server-ip>
  ProxyCommand none

Host mon
  User <admin-username>
  Hostname <monitor-server-ip>
  ProxyCommand none
```

Finally, test direct SSH access from the terminal, using the commands `ssh app` and `ssh mon`. It should be possible to connect without entering a password.

1.60.4 Step 4: Retrieve SecureDrop configuration info from the servers

In addition to the account and networking information retrieved from the servers so far, you'll need to retrieve the following files and info:

- GPG *Submission Public Key*, *OSSEC Alert Public Key*, and (optional) *Journalist Alert Public Key*
- OSSEC alert configuration details
- (Optional) HTTPS configuration details

Retrieve GPG public keys

Copy the *Submission Public Key* with the following commands:

```
echo "$(ssh app sudo cat /var/lib/tor/services/sourcev3/hostname)" > /tmp/sourcev3
cd ~/.config/securedrop-admin
curl http://$(cat /tmp/sourcev3)/public-key > SecureDrop.asc
gpg --import SecureDrop.asc
```

Validate that the imported key's fingerprint matches the one on your SecureDrop install. You can do this by running the command:

```
gpg --with-fingerprint --import-options import-show --dry-run --import SecureDrop.asc
```

Then, compare the returned fingerprint value with that advertised by your *Source Interface*, using the command:

```
curl http://$(cat /tmp/sourcev3)/metadata
```

Next, note the OSSEC Alerts email address (`OSSEC_EMAIL`) and, if applicable, the Daily Journalist Alerts email address (`JOURNALIST_EMAIL`):

```
ssh mon sudo cat /var/ossec/send_encrypted_alarm.sh | grep _EMAIL= | cut -f7 -d' '
```

Import the *OSSEC Alert Public Key* using the following commands (substituting the appropriate email address for `alerts@example.com`):

```
ssh mon sudo gpg --homedir=/var/ossec/.gnupg --export --armor alerts@example.com > ossec.
↪pub
gpg --import ossec.pub
```

If a Daily Journalist Alerts address has been configured, repeat this step for the *Journalist Alert Public Key*, naming it `journalist.pub` or similar.

You will require the fingerprints for these keys during the next step, which you can obtain via the command:

```
gpg -k --fingerprint
```

Retrieve OSSEC alert configuration details

You'll also need to retrieve the following configuration information:

- SMTP server
- SMTP port
- SASL username
- SASL domain
- SASL password

To retrieve these values, use the following command in the terminal:

```
ssh mon sudo cat /etc/postfix/sasl_passwd
```

This will return a line like:

```
[smtp.gmail.com]:587 testossec@gmail.com:AwfulPassword
```

In this example, `smtp.gmail.com` is the SMTP server, `587` is the SMTP port, `testossec` is the SASL username, `gmail.com` is the SASL domain, and `AwfulPassword` is the SASL password.

(Optional) Retrieve HTTPS certificate files

If your *Source Interface* was configured to use HTTPS, you will need to copy three related files from the *Application Server* to the *Admin Workstation*.

To retrieve these files, use the commands:

```
cd ~/.config/securedrop-admin
ssh app sudo tar -c -C /var/lib ssl/  | tar xvf -
```

These commands will create a directory named `~/.config/securedrop-admin/ssl` on the *Admin Workstation*, containing your instance's SSL certificate, certificate key, and chain file. When prompted for the names of these files during the next step, you should specify them relative to the `~/.config/securedrop-admin/` directory, i.e. as `ssl/mydomain.crt`.

1.60.5 Step 5: Configure and back up the *Application Server*

Next, configure the SecureDrop *Application Server* using the files and info retrieved in the previous steps. To do so, connect to the Tor network on the *Admin Workstation*, open a Terminal and run the following commands:

```
securedrop-admin sdconfig
```

The `sdconfig` command will prompt you to fill in configuration details about your instance. Use the information retrieved in the previous steps. When prompted whether or not to enable SSH-over-Tor, type **no**.

Next, back up the *Application Server* by running the following command in the terminal:

```
securedrop-admin backup
```

Ensure the backup command completes successfully.

1.60.6 Step 6: Use the installer to complete the configuration

Run:

```
securedrop-admin install
```

Once the command completes successfully, run

```
securedrop-admin localconfig
```

Once this command is complete:

- verify that the *SecureDrop Menu* for the *Source* and *Journalist Interfaces* works correctly, opening their respective homepages in Tor Browser.

To revert the changes made to enable temporary local SSH access, you should reboot the servers, by issuing the following commands in a terminal:

```
ssh app sudo reboot
ssh mon sudo reboot
```

1.60.7 Step 7: Set up SSH-over Tor

Note

Without performing this step, you will not be able to access your SecureDrop servers from outside the local network. See *SSH over local network* for more information.

Rerun the command:

```
securedrop-admin sdconfig
```

Press "Enter" to use the pre-populated values, but when asked whether to configure SSH-over-Tor, type **yes** (recommended).

Then, re-run

```
securedrop-admin install
```

When the installation completes, run:

```
securedrop-admin localconfig
```

Once this command completes:

- verify that the Hostname references in `~/.ssh/config` have been updated to refer to onion addresses instead of direct IP addresses
- verify that you can connect to the servers using `ssh app` and `ssh mon`
- verify that the *SecureDrop Menu* for the *Source* and *Journalist Interfaces* works correctly, opening their respective homepages in Tor Browser.

1.60.8 Step 8: Post-rebuild tasks

Important

Rebuilding an *Admin Workstation* makes changes that will prevent your other Tails workstations from connecting to your SecureDrop servers. If you rebuild your *Admin Workstation*, you must also provision all other existing Tails workstation drives updated Tor credentials (see below).

We recommend completing the following tasks after the rebuild:

- Set up a new administration account on the *Journalist Interface*, by following *these instructions*
- Verify that submissions can be decrypted, by going through the decryption workflow with a new submission.
- Back up your *Admin Workstation*.
- Delete invalid admin accounts in the *Journalist Interface*.
- Restrict SSH access to the *Application* and *Monitor Servers* to valid *Admin Workstations*. If your new *Admin Workstation* USB flash drive is the only one that should have SSH access to the servers, you can remove access for any previous *Admin Workstations* from the terminal, using the commands:

  ```
  securedrop-admin reset_admin_access
  ```

  You can also selectively remove invalid keys by logging on to the *Application* and *Monitor Servers* and editing the file `~/.ssh/authorized_keys`, making sure not to remove the public key belonging to your new *Admin Workstation*.
- *Back up the *Application Server** once SSH-over-Tor has been restored. Ensure that server and workstation backups happen regularly.
- Provision all other Tails workstation drives (*Journalist* and/or *Admin Workstations*) with updated Tor credentials, so that they can access SecureDrop after this rebuild.

  You will need to copy the following file(s) to all other *Admin* and *Journalist Workstations*, replacing the existing files of the same name:

  ```
  ~/.config/securedrop-admin/app-journalist.auth_private
  ~/.config/securedrop-admin/tor-v3-keys.json # for Admin Workstations only
  ```

  You may copy these files using a *Transfer Device* (which must be wiped afterwards), or boot into each of your additional Tails workstations, plug in and unlock your *Admin Workstation*'s encrypted partition via the **Places** app, and manually copy the file(s) from the *Admin Workstation* to the same directory on the target Tails workstation.

1.61 Updates over Tor

In case of censorship or blocking of the SecureDrop APT repository (`apt.freedom.press`), which provides automatic updates, Tor can be configured to provide unrestricted access.

Note

This is only meant as a temporary measure. SecureDrop generally expects an unfiltered internet connection. If you are facing long-term censorship, *please contact us* for other options.

1.61.1 Configuring updates over Tor

These steps will need to be applied to both the *Application Server* and the *Monitor Server*.

As mentioned earlier, this is meant to be a temporary measure. Notably, running `securedrop-admin install` will overwrite these changes.

1. From your *Admin Workstation*, SSH into the *Application Server* or *Monitor Server* using `ssh app` or `ssh mon`.
2. Run `sudo nano /etc/tor/torrc` to edit the Tor configuration. Replace the first line of `SocksPort 0` with `SocksPort 127.0.0.1:9050` and save the file.
3. Run `sudo systemctl reload tor@default` for the new configuration to take effect.
4. Run `sudo apt-get install apt-transport-tor --yes`.
5. Run `sudo nano /etc/apt/sources.list.d/apt_freedom_press.list` to edit the URL to begin with a "tor+" prefix. The new contents should be:

```
deb [arch=amd64] tor+https://apt.freedom.press noble main
```

6. Run `sudo apt update` and verify there are no error messages. This checks that fetching updates works

1.61.2 Disabling updates over Tor

From your *Admin Workstation*, run `securedrop-admin install`. This will overwrite all the above changes.

1.62 Troubleshooting kernel updates

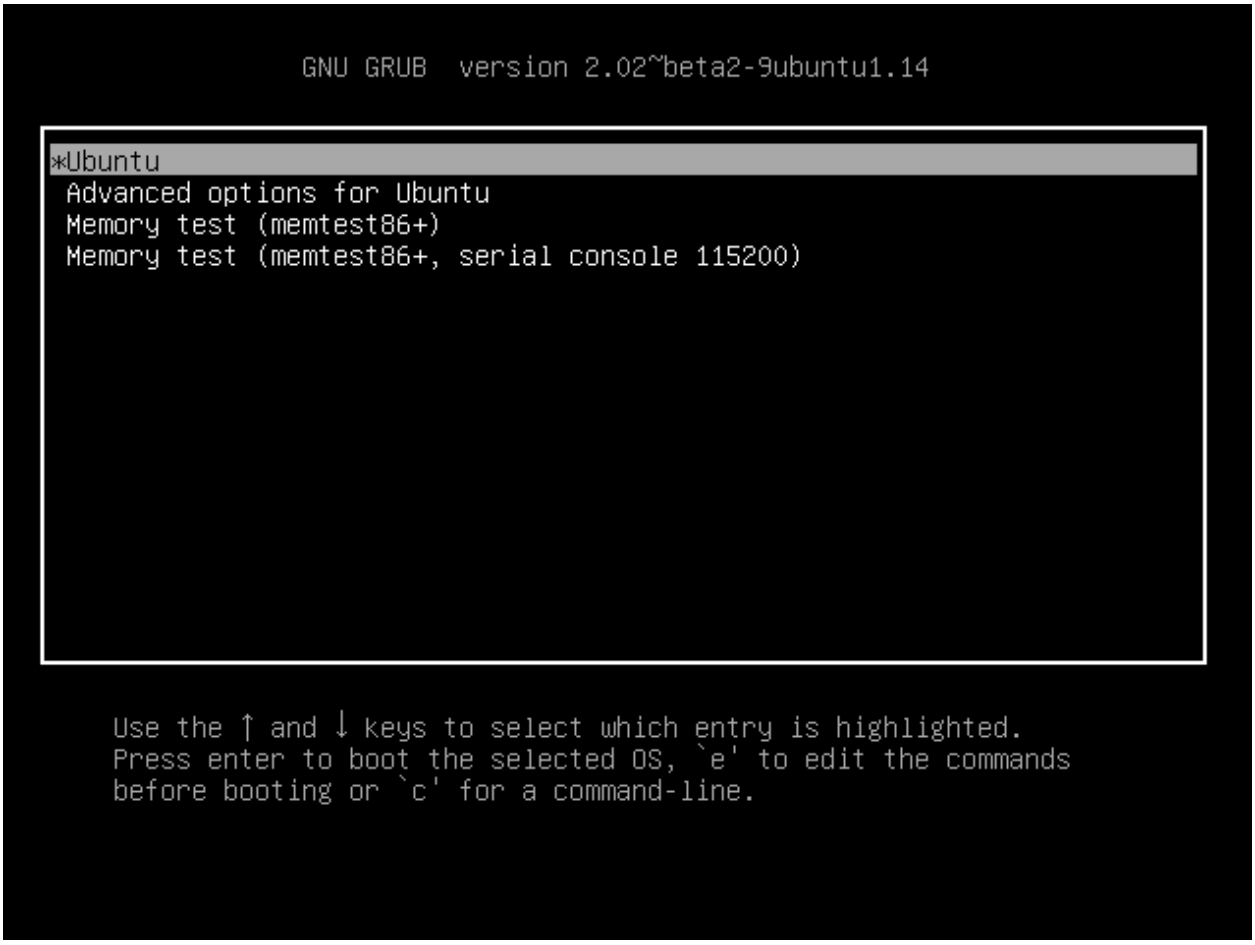
Kernel updates address known bugs and security vulnerabilities in the Linux kernel. They may be installed automatically on your *Application* and *Monitor Servers* as part of a SecureDrop release. All kernel updates are tested extensively against *recommended hardware*. If things do go wrong (e.g., the server does not boot after a kernel update), the following instructions will help you to roll back to the previous, working kernel. You can then *report compatibility issues* to us so we can work together to resolve them as quickly as possible.

First, you need to physically access each server. Power down the server (safely if possible), attach required peripherals (keyboard, monitor), and power the server back up.

If you have access to the password for your admin user, you can use it to log into each server without the use of *Two-Factor Authentication*, which was disabled for keyboard logins in SecureDrop 0.8.0. You may have saved the password in the KeePassXC database on your *Admin Workstation*. If you do not have the password, you can boot into single user mode instead.

1.62.1 Boot into single user mode

To access single user mode, you will have to edit the boot options for the new kernel. You can do so using the GRUB bootloader, pictured below:



Press any key quickly just once. You will only have about 2 to 3 seconds before Ubuntu starts booting. If you miss that window, just log in normally and reboot safely, provided you can log in. Do not unplug or forcibly shut down the server.

Once you hit a key, you will be able to interact with the menu with the up (↑) and down (↓) keys. Select "Ubuntu" as shown above, and press "e" to edit the boot options. In the line that begins with "linux", add the word "single" at the end. When you are done, the output on your console should look similar to the screenshot below.

Press the "F10" key to boot.

1.62.2 Test the new kernel

Observe the boot process. It is possible that the system will fail to boot completely; if so, the log information will help us to understand what is happening.

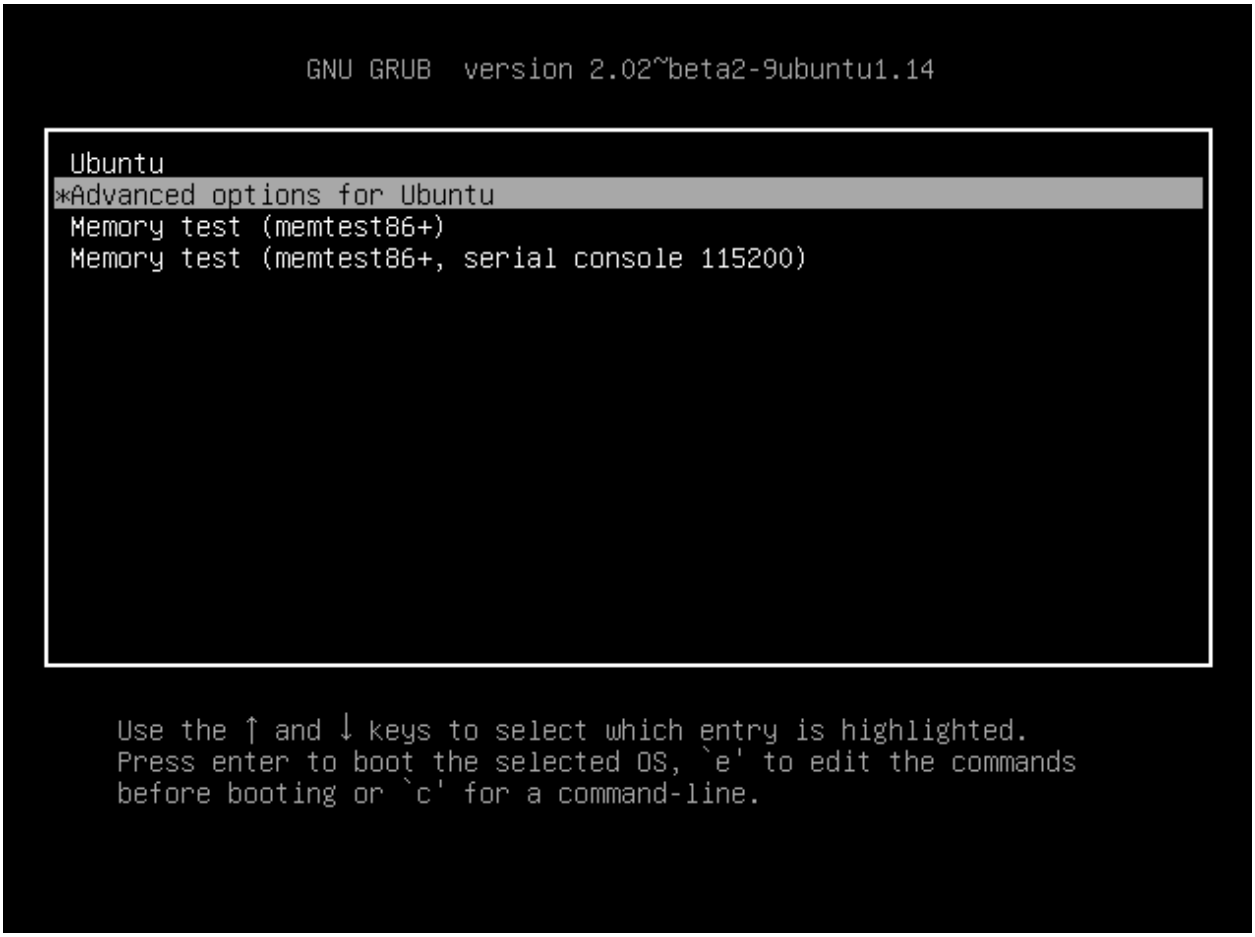
Provided that you can log in, check if you have network access. Try a command such as `sudo host freedom.press`. If you don't have network access, it is most likely due to the upgraded kernel missing a network driver for your hardware.

If everything appears to be operating normally, the outage may not be kernel-related. In that case, you may still wish to follow the steps at the end of this document to send us log information along with an issue report, and we will help you investigate.

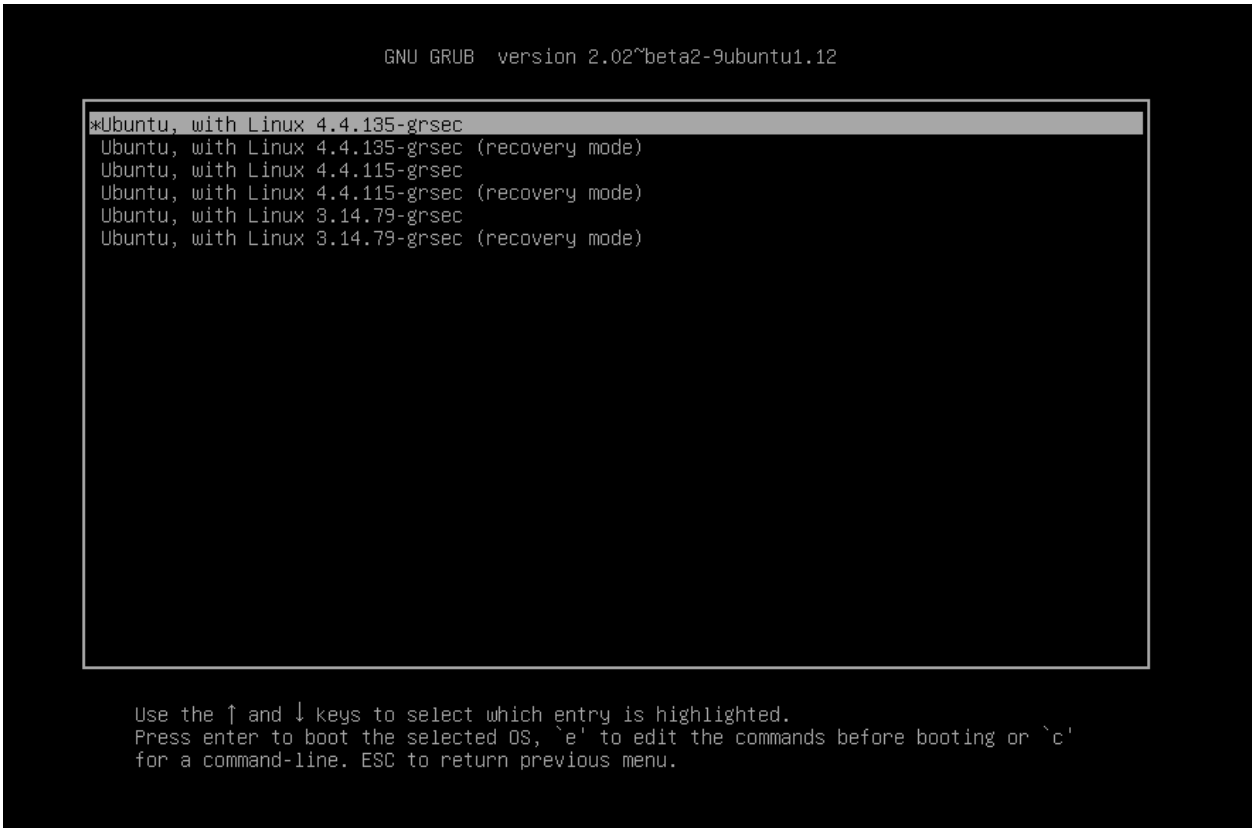
If you are experiencing network issues or other kernel problems, we recommend that you roll back to an older kernel, and that you report the issue to us immediately.

1.62.3 Compare the behavior of the old kernel

Reboot the server in a safe way with `sudo reboot`. After the BIOS screen, you can select a different kernel from the GRUB boot menu by selecting **Advanced options for Ubuntu**, pictured below.



The next menu should give you a list of kernels, similar to the one pictured below:



Choose the option with the previous kernel version. If unsure, please consult the release notes for the most recent release of SecureDrop, which will include details about kernel version changes.

As before, you may need to edit the kernel options to enter single user mode. The boot process should proceed normally. Wait until you get a login prompt and log in.

Once you are logged in, check to see if you have network access. If you do, then your instance is having an issue with the newer kernel. In that case, we need to temporarily set an older kernel as the default.

1.62.4 Roll back to the old kernel

Important

It is of critical importance for the security of your instance that we work together to resolve any compatibility issues. Rolling back to an older version is only a stopgap measure to avoid a prolonged outage of your SecureDrop instance.

Inspect the file `/boot/grub/grub.cfg`. You should find a `menuentry` line with the same text that you selected during boot, e.g.:

```
submenu 'Advanced options for Ubuntu'...

  menuentry 'Ubuntu, with Linux 4.xxx.xx-grsec...
```

Take note of its position among the other submenu entries (it will most likely be third). Then edit the GRUB configuration:

```
sudo nano /etc/default/grub
```

Make a backup of the file or take a note of the current value of `GRUB_DEFAULT` somewhere, so you can restore the previous behavior easily at a later point.

Once you have done so, set the `GRUB_DEFAULT` variable to point to the index of the menu and submenu. Note that the index starts at 0, so for a typical setup, the line in `/etc/default/grub` would look like this:

```
GRUB_DEFAULT="1>2"
```

The "1" means the second entry of the main menu ("Advanced options"), the "2" means the third entry of the submenu. Again, update these numbers consistent with your configuration.

Caution

Ensure that you have chosen the right index for the main menu and the submenu, and double-check that you are beginning the count at 0, not 1; otherwise, you may boot into the wrong kernel.

This change still has to be applied to take effect on the next boot:

```
sudo update-grub
```

Now you can reboot into the old, working kernel.

```
sudo reboot
```

The server should come up automatically. From here on, you should be able to perform all administrative tasks via SSH as usual. If you want additional confirmation of the kernel version, the command `uname -r` should display the expected kernel version number.

Please notify us of the compatibility issue so we can help you resolve it ASAP.

1.62.5 Report compatibility issues

If you have encountered issues with a kernel update, it is important that you report them to us so that we may incorporate any necessary changes to our updated kernel, and so that we can work with you to switch back to the new kernel as soon as possible.

Run the following commands via SSH from the *Admin Workstation*:

```
source /usr/share/securedrop-admin/venv/bin/activate
cd /usr/share/securedrop-admin/ansible-base
ansible all -b -m setup > ~/server-facts.log
```

Please also send us a copy of `/var/log/syslog` and `/var/log/dmesg` for analysis.

You can share `server-facts.log`, `syslog` and `dmesg` with us as follows:

- If you are a member of our Support Portal, please create a new issue and attach the files to it.
- Alternatively, email us at securedrop@freedom.press (GPG encrypted) with the subject "SecureDrop kernel facts" and the files attached.

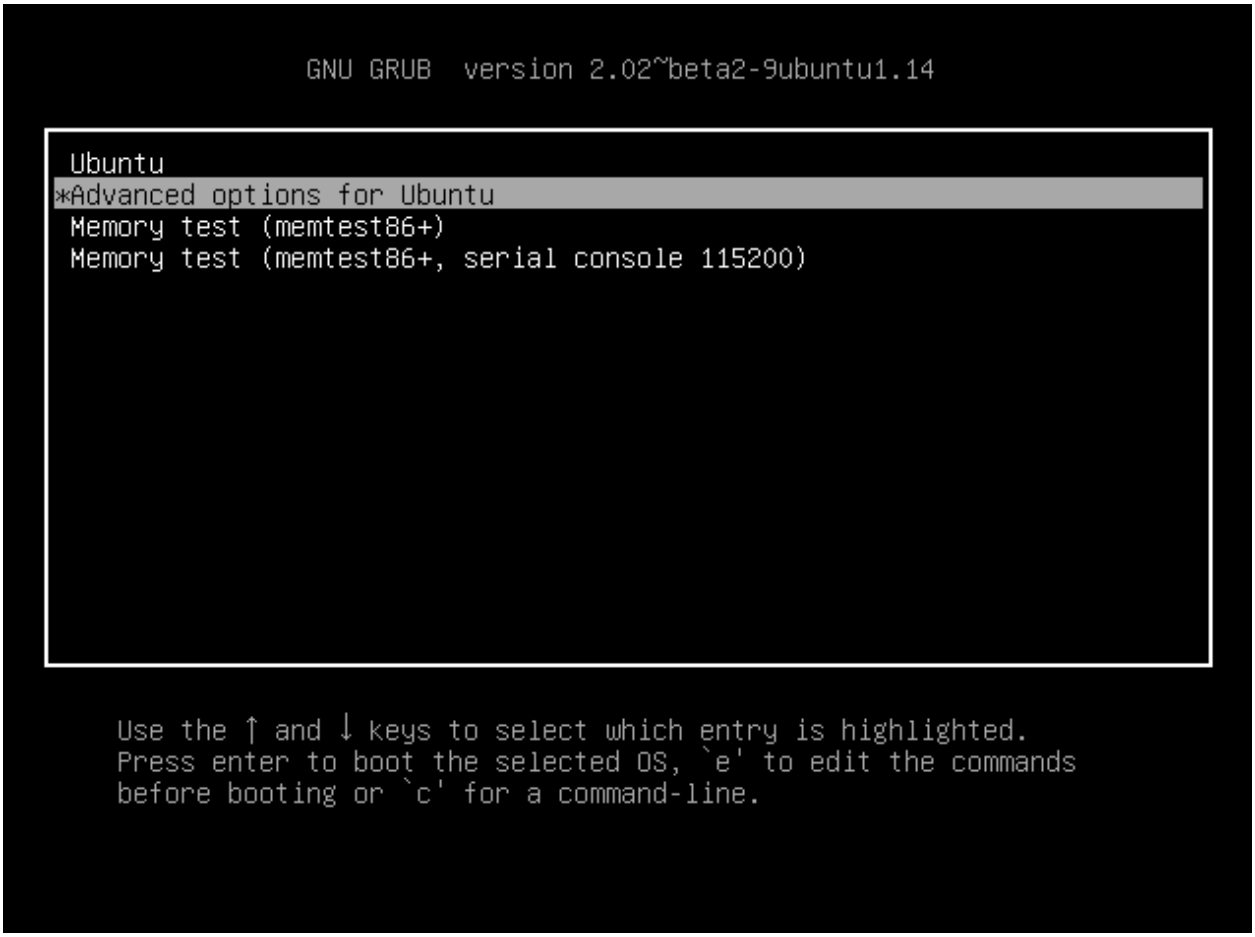
Once we get your information, we can try to provide assistance to resolve compatibility issues.

1.62.6 Test and enable an updated kernel

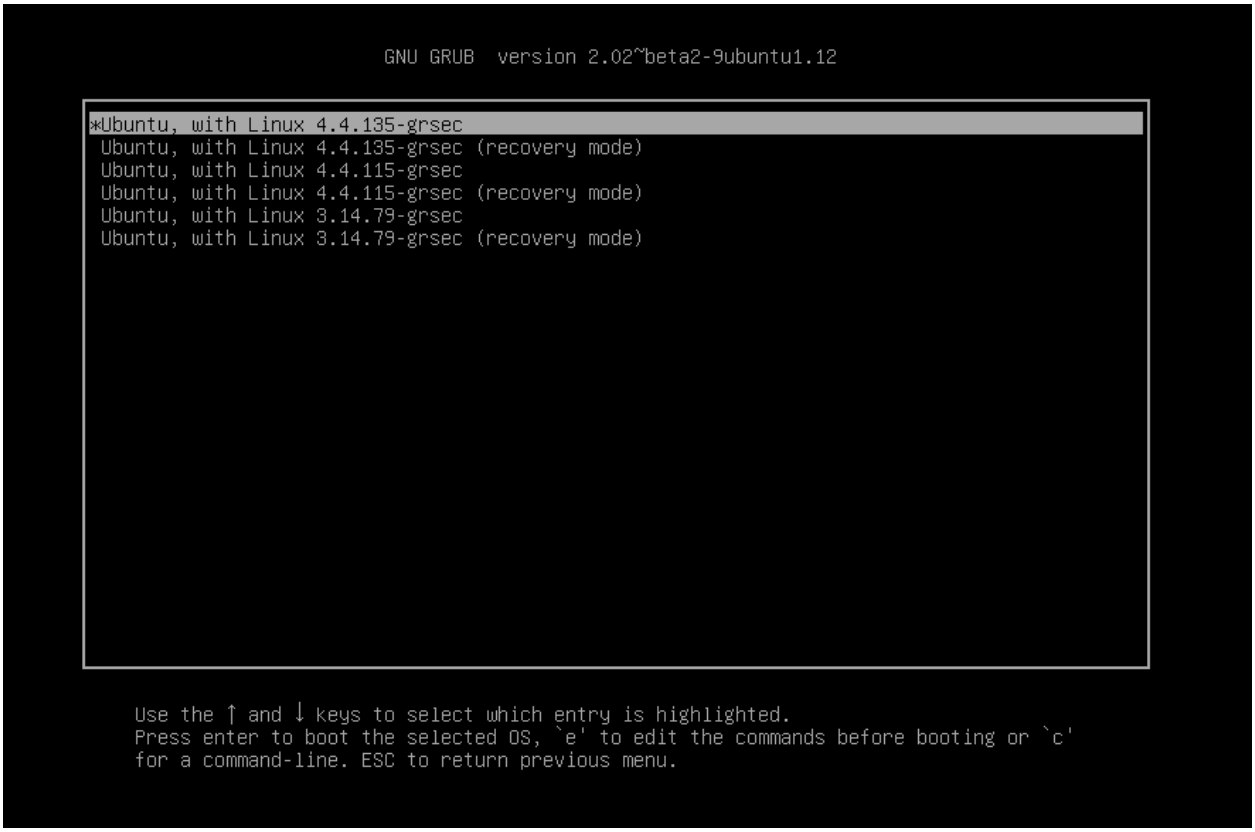
If you have changed your default kernel, we urge you to test an updated kernel as soon as it becomes available in a future SecureDrop release. Note that an update may be enforced as part of a release to protect the security of your instance. Please consult the release notes for details about kernel updates.

You can test a kernel update without downtime for your instance by booting your *Monitor Server* with the new kernel. Log into your *Monitor Server* using the *Admin Workstation*. Shut down the server safely using the command `sudo poweroff`. Ensure that the server is fully powered off.

Attach required peripherals and power the server back up. After the GRUB bootloader appears, select **Advanced options for Ubuntu**, pictured below.



If a SecureDrop release with a kernel update has been installed on your system, the updated kernel version will be available in the list of options:



Select the new kernel (you do not need to use the version with recovery mode). If you do not know your admin account password, you can *boot into single user mode* by editing the boot options. Otherwise, press enter to boot.

Verify that you can boot successfully, and that you have network access (`sudo host freedom.press`). If you still encounter problems with the new kernel, please *report compatibility issues* at your earliest convenience, and reboot the server into the old kernel for now.

If the update resolved compatibility issues with an earlier kernel version, you can make the new kernel the default. Edit the file `/etc/default/grub`, e.g., by issuing the following command:

```
sudo nano /etc/default/grub
```

Make a backup of the file or take a note of the current value of `GRUB_DEFAULT` somewhere, so you can restore the previous behavior if needed. Change the line to `GRUB_DEFAULT=0`. This configures the bootloader to default to loading the most recent kernel version installed on your server.

This change still has to be applied to take effect on the next boot:

```
sudo update-grub
```

Safely shut down the *Monitor Server*, remove attached peripherals, and reboot it. Verify that it is working correctly by logging in using your *Admin Workstation*. If everything is working as expected, you can make the same change to `/etc/default/grub` on your *Application Server* as well. Remember to again run the command `sudo update-grub` when you are done.

You can make the change on the *Application Server* from your *Admin Workstation* and reboot the server using the command `sudo reboot`.

Subsequent kernel updates will again be applied automatically.

1.63 BIOS updates on the servers

Below are the steps for updating the BIOS on the *Application* and *Monitor Servers*. We provide instructions for Intel and ASUS NUC devices, in accordance with *our hardware recommendations*. You should also update the BIOS on other computers such as the *Admin Workstation,* but those instructions will vary depending on the manufacturer and model of your device.

1.63.1 What you need

1. A clean USB flash drive to download the BIOS file
2. An Internet-connected workstation, such as the *Admin Workstation*
3. A UPS (uninterrupted power supply), such as a surge-protecting power supply with a backup battery (This is not required, but strongly recommended)
4. A keyboard and monitor

1.63.2 Perform backups

If you are updating the BIOS on an existing SecureDrop system, we recommend you *back up the *Application Server** before proceeding.

1.63.3 Prepare the USB flash drive

Using the Disks application, delete existing partitions on the USB flash drive, if applicable, and reformat the entire device with one FAT32 partition. Note that you will lose access to all existing data on this USB flash drive.

1.63.4 Download and verify appropriate BIOS files

For Intel and ASUS NUC devices

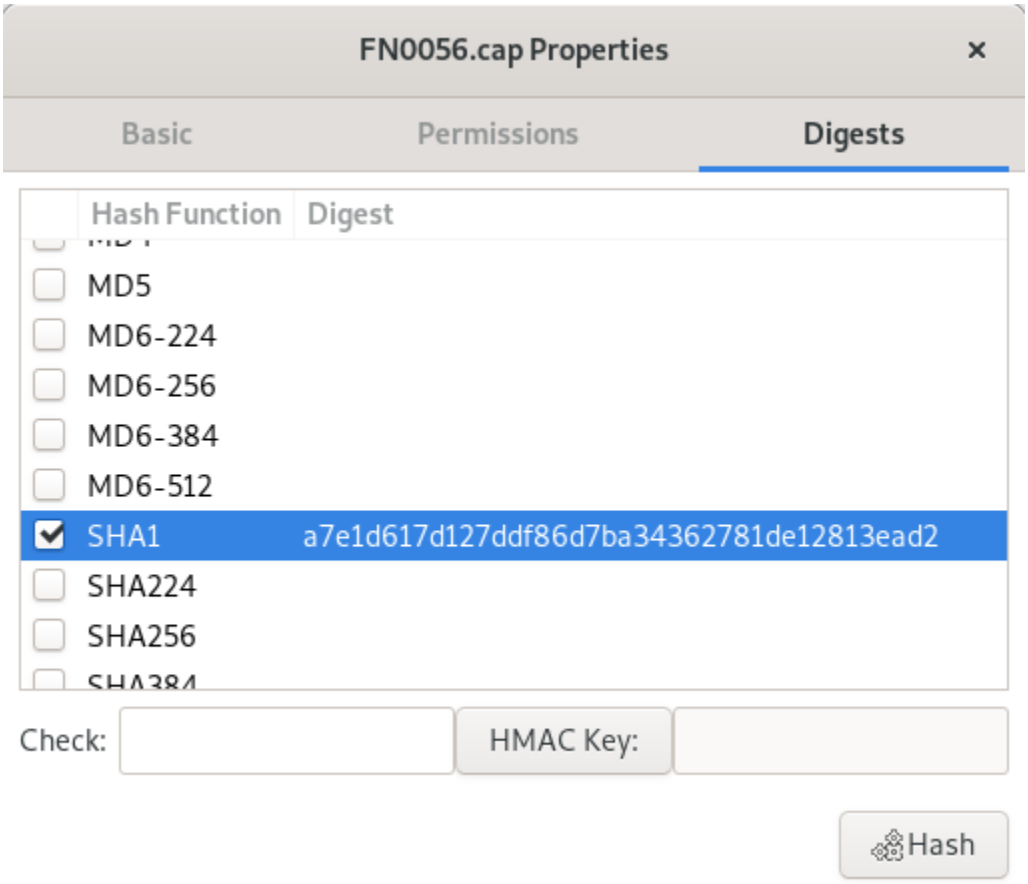
Check the make and model of your servers, and follow the F7 BIOS update method in the documentation. The exact instructions vary by model:

- BIOS update instructions for Intel NUC with Intel Visual BIOS
- BIOS update instructions for Intel NUC with Aptio V UEFI Firmware Core
- BIOS update instructions for ASUS NUCs

Each make and model of NUC will offer different file types; proceed to either the Intel or ASUS Download Center and download the file indicated in the documentation for the F7 method (e.g., `.bio` or `.cap`).

> **Warning**
>
> Do not download BIOS updates from anywhere other than the manufacturer's website. Be sure that you are on the correct website and that it has a valid SSL Certificate. Intel's SSL Certificate is issued to *.intel.com and signed by DigiCert. ASUS' SSL Certificate is issued to *.asus.com and signed by Amazon. Be sure you download the files specific to the model of your servers.

Intel provides an SHA1 checksum on the download page, while ASUS offers a SHA-256 checksum. Once you have downloaded the file, using the **Files** application, browse to the file, right click and select **Properties ▶ Digests**, select either SHA1 or SHA256 depending on which is available to you, and click Hash. Compare the result in the Digest column to the checksum listed on the manufacturer's website. If these two values do not match, do not proceed, and contact support@freedom.press. Tails provides a detailed explanation of this process. (Note that the hash in the screenshot below is an example only, and will not match your specific file.)



Once you have verified the hash, copy the file to your USB flash drive.

1.63.5 Update the BIOS

Power off the *Monitor Server*. We recommend plugging it into an uninterrupted power supply (UPS). Plug in the keyboard, monitor, and USB flash drive, and power on the server, then press F7 when prompted to enter the BIOS Update tool.

Select the USB flash drive and navigate to the file you have downloaded, then hit **Enter**. The update will take several minutes–do not interrupt the update or unplug the server during this time.

Repeat these steps on the *Application Server*.

1.64 Decommission SecureDrop

1.64.1 Protecting, moving, or taking down your SecureDrop instance

If the location hosting your SecureDrop servers is going to be empty for extended periods of time, you should take steps to ensure the security of your servers and associated hardware:

1. Ensure that the room where the servers are installed is locked by default, and that only authorized personnel have access. If possible, have access logged.
2. If the server room is covered by CCTV, verify that the footage will be monitored or reviewed periodically.
3. Ask to have adjacent corridors included in any regular security patrols.

4. Ask *Journalists* to purge old submissions, to reduce the impact if the servers are compromised (this is good general practice in any case).
5. If your SecureDrop instance is set up to allow SSH-over-LAN admin access, consider switching it to SSH-over-Tor access instead. To do so, you will need to update the server configuration using the Admin VM.

In some cases, if you are not able to ensure the security of your instance during periods of prolonged absence, it may be better to relocate it, or in extreme circumstances, temporarily take it down. If you decide to take down your SecureDrop instance, we recommend the following steps:

1. Consult with *Journalists* using the system, to ensure that any active *Sources* are aware of the situation, and that source conversations can either be paused or continued via other means.
2. Update your SecureDrop *Landing Page* (typically a "send us tips" page, or a page linked from there) to let prospective *Sources* know that the outage is coming, and optionally to redirect them to other contact methods, such as a shared Signal tipline.
3. *Back up your servers*.
4. Power down the servers, and remove them and the network firewall from the server room. Store the equipment securely offsite.

Warning

By default the SecureDrop servers are not set up with full disk encryption enabled, to allow for hands-off reboots. This means that it is crucial that they be kept secure. If the servers are lost or stolen, an adversary would gain access to all encrypted submissions and messages. While they would not be able to decrypt them, this would still provide valuable metadata about source conversations.

In most cases, restoring the instance, whether in their original hosting location or elsewhere, is a matter of reconnecting the servers to the firewall, attaching a WAN connection that allows unfiltered access to Tor to the firewall WAN port, and powering everything on.

1.64.2 Permanently decommissioning SecureDrop

The following steps will guide you through the decommissioning of your SecureDrop instance.

1. **Put a notice in advance on your *Landing Page* to inform sources that your instance will soon be retired.** You may want to direct them to other secure methods of contacting you.
2. **Locate and create an inventory of all your hardware.**
    - *SecureDrop Workstation* laptops
    - *Export Devices* (USB flash drives, optical drives, or external drives)
    - Backup USB flash drives/other storage media
    - Servers
    - Firewall

    You may also want to inventory credentials, such as the email address or alias and PGP key used for receiving OSSEC alerts, in order to retire them.
3. **Optional: Save a backup.** If you want to save a backup of the *Application Server* (for example, to reinstall SecureDrop in the future using the same onion address), follow our *backup guidelines*. Once the backup has been created, you can move it onto an encrypted drive, such as a LUKS-encrypted USB flash drive. You will also require a backup of the *Submission Private Key* found on the *SecureDrop Workstation*.

If you do not require a server backup, you may choose to download specific submissions, and store them in a secure manner (such as on an encrypted USB flash drive).

4. **Optional: Delete submissions on the server.** Log into the *Journalist Workstation* and delete all sources to take advantage of SecureDrop's secure deletion properties. Note that depending on the number of sources on your server, it may take anywhere from several minutes to an hour or more for the submissions to be completely deleted from the server.

   You can either leave the server ample time to complete this operation, or monitor the progress by SSHing to the *Application Server* and running

   ```
   sudo journalctl -f
   ```

   You will see repeated log lines that contain the following:

   ```
   [Timestamp] app python [...] INFO Clearing shredder
   [Timestamp] app python [...] INFO Files to delete: <number>
   ```

   When the number of files to delete reaches 0, the process is complete.

5. **Disconnect the firewall and the servers from the internet.** Be sure to inform your network administrator of any changes to devices on your network.

6. **Wipe and destroy the USB flash drives.** Because the USB flash drives used for SecureDrop are all LUKS-encrypted, reformatting the USB flash drives (in particular, overwriting a portion of internal storage called the **LUKS header**) should be sufficient to make any existing data on those drives unrecoverable.

   For example, you could use Tails to launch Gnome Disks, insert and identify the USB flash drive you are trying to erase, and reformat this drive with a new, LUKS-encrypted partition, erasing the existing partition data.

   Caution

   Be **very** sure you are reformatting the right drive. You may want to use the *Secure Viewing Station* laptop for this procedure to reduce the risk of accidentally erasing a drive on your regular-use machine.

   You may also choose to destroy the drives by physical means, such as using a hammer or purpose-built shredder to pulverize or destroy the drive.

7. **Wipe and destroy the storage drives on the servers.** SecureDrop submissions are stored GPG-encrypted on the *Application Server*. Unless your SecureDrop *Submission Key* is compromised (or a significant vulnerability in GPG is discovered), access to the servers does not guarantee access to the submissions and messages you have received.

   That said, there may still be some sensitive information on the servers, including system logs and the SecureDrop database, which would yield information on the number of submissions and replies stored on the server. This risk is partially mitigated by securely deleting submissions from the server, as described in a previous step; however, physically destroying or encrypting the storage drives on the servers are the best ways to ensure that data on the drives cannot be recovered.

   Physically destroying SSD drives is not as straightforward as destroying older hard drives, but drives can be pulverized, shredded, or incinerated, as long as the flash chips are destroyed.

   If those options are not available, you may choose instead to write over the information on the existing drives. Most SSDs support ATA Secure Erase, although the implementation of this feature varies by manufacturer.

   Another option is to re-install a clean version of Ubuntu server with full- disk encryption enabled. During the disk-partitioning portion of the installation wizard, select *Guided - use entire disk and set up encrypted LVM*.

You will need to reclaim the space that was taken up by your previous installation, so whenever prompted to unmount and reclaim unused partitions, select "yes."

8. **Destroy Export media, if applicable.**
9. **Optional: Factory-reset the firewall.**
10. **Update your *Landing Page* (tips page) to reflect the fact that your organization no longer has SecureDrop.**
11. **Notify the SecureDrop Support team that your instance is no longer active.** If you have any questions about the decommissioning process, or about other secure communications options, please feel free to contact us at securedrop@freedom.press (GPG encrypted) or via the support portal.

1.65 Backup and restore

Qubes OS has a backup utility that allows for backup and restoration of user-specified VMs and templates.

SecureDrop Workstation requires only that you back up instance-specific secrets and configuration files, although you can optionally back up some additional local data.

To perform backups, you will need:

- a LUKS-encrypted USB or LUKS-encrypted external hard drive (of sufficient size, if backing up additional local data)
- a secure place to store backup credentials (such as a password manager on your primary laptop)

1.65.1 Backup

Preserve files from `dom0` and `sd-gpg`

Preserve configuration files and private key material by copying them into `dom0`.

In a `dom0` terminal opened via  ►  ► **Other Tools ► Xfce Terminal**:

```
qvm-run --pass-io sd-gpg 'gpg -a --export-secret-keys' > sd-keys.asc
sudo mv sd-keys.asc /usr/share/securedrop-workstation-dom0-config/
cp -r /usr/share/securedrop-workstation-dom0-config ~
```

If you have made customizations to `dom0` (for example, custom RPC policy files):

```
mkdir ~/etc-qubes && cp -r /etc/qubes ~/etc-qubes
mkdir ~/etc-qubes-rpc && cp -r /etc/qubes-rpc ~/etc-qubes-rpc
```

Back up a SecureDrop Workstation

Note

Backups contain sensitive data, and must be created and stored just as securely as SecureDrop Workstation itself.

If performing this backup as part of a migration (from one machine to another or from one version of Qubes OS to another), we suggest you retain the backup only during the migration process, and destroy it after the migration is complete. The easiest way to do this is to create a LUKS-encrypted drive, follow this guide to create your backup, and then wipe (reformat) or destroy the drive after you have successfully restored it onto the new machine, which should ideally happen the same day. In all cases, follow your organization's internal policies on handling sensitive assets and information.

If you are looking to back up your own customized components of SecureDrop Workstation for long-term storage, we suggest taking that backup separately from the backup of SecureDrop Workstation components so that you can avoid proliferating copies of sensitive assets.

Before starting your backup, decide whether you want to back up your data from `sd-app`. If you skip this step, the first time you log in, your submissions will re-download from your SecureDrop server.

Ensure your storage medium is plugged in, attached to `sd-devices`, and unlocked.

Navigate to  ►  ► **Qubes Tools** ► **Backup Qubes**, and move all VMs from "Selected" to "Available" by pressing the << button.

To target a VM for backup, highlight it and move it into the "Selected" column by pressing the > button. Select:

- `dom0`
- the `sd-app` VM (optional), noting the warning above
- any customized VMs (and their templates) that you may wish to preserve, noting the warning above.

You do not need to back up the other `sd-` VMs.

Click "Next", and in "Backup destination," specify the VM and directory corresponding to your storage medium's current mount point.

Set a strong, unique backup passphrase (7-word diceware), and ensure this passphrase is stored securely outside SecureDrop Workstation.

Note

This passphrase protects sensitive components of your SecureDrop instance, including the *Submission Private Key*, and unencrypted submissions (if `sd-app` is backed up). Ensure it is a very strong password and is stored securely.

Uncheck "save backup profile," then proceed with the backup.

Qubes OS recommends verifying the integrity of the backup once the backup completes, and this should be done on the same machine where the backup was created. This can be done by using the Restore Backup GUI tool and selecting "Verify backup integrity, but do not restore the data." For details, see the Qubes OS backup documentation.

Warning

Any files or data not mentioned above and not backed up elsewhere will be destroyed. Ensure that any other data on your system (for example, using KeepassXC in the `vault` VM, or data stored in other VMs) have been backed up and the integrity of the backup has been verified before proceeding.

1.65.2 Restore

Reinstall Qubes OS

To restore SecureDrop Workstation, follow our *pre-install tasks* to provision a Qubes OS system complete with updated base templates.

Rename or delete redundant AppVMs

By default, Qubes OS will create the AppVMs `personal`, `work`, `untrusted` and `vault` as part of the installation process. Rename or delete any of these newly created AppVMs whose names conflict with the AppVMs you intend to restore from a backup.

Example: If you wish to restore the `vault` VM, rename or delete the existing `vault` VM prior to restoring the backup.

You can do so in ► **Apps** ► **vault** ► **Settings** (the VM must not be running).

Restore backup (SecureDrop Workstation components)

Plug in your backup medium and unlock it as during the backup. By default on a new system, your peripheral devices will be managed by a VM called `sys-usb`.

Navigate to ► ► **Qubes Tools** ► **Restore Backup**, and enter the location of the backup file. You do not need to adjust the default Restore options, unless you have made customizations to the backup. Enter the decryption/verification passphrase, and proceed to restoring the available qubes (which should include `dom0` and possibly `sd-app`).

We suggest restoring only those VMs, provisioning SecureDrop Workstation, and then restoring any customized VMs you may have had once that process is complete. This way SecureDrop Workstation is provisioned on a clean system and can implement the security measures it requires before any additional VMs are configured.

Note

When migrating to a newer version of Qubes OS (for example, Qubes 4.1 to Qubes 4.2), you may notice that the original templates for certain VMs are not present on your new machine. For the purposes of this guide (optional `sd-app` backup), this is not a problem. Allow the VM to be restored with the default template suggested by the operating system (the current Fedora base template). **Do not start the VM.** Continue through the reinstallation process. The correct template will be configured as you follow the rest of these instructions.

If you are restoring your own customized VMs and templates, you will need to take additional steps. You may decide to create new templates for your custom VMs and provision them with the necessary applications/customizations (recommended), or you may upgrade your existing templates following the upstream documentation (Fedora templates, Debian templates), then upgrade their package repositories to the Qubes 4.2 repositories using:

```
sudo qubes-dom0-update -y qubes-dist-upgrade
qubes-dist-upgrade --template-standalone --upgrade
```

More information can be found in the upstream documentation. Contact Support with any questions.

Reinstall SecureDrop Workstation

If you do not already have a `work` VM, create it with default networking settings:

```
qvm-create -l blue work
```

Then, *download and verify* the SecureDrop Workstation .rpm to the `work` VM and copy it to `dom0`.

Once you have a valid .rpm file in `dom0`, install the .rpm by running:

```
sudo dnf install securedrop-workstation.rpm
```

Retrieve the previous SecureDrop Workstation configuration from the backup folder on `dom0`. From the `dom0` home directory:

```
ls -d */*/* | grep home-restore
```

You should see a directory called `home-restore-$YYYY-MM-DD-HHMMSS/dom0-home/$USERNAME`. We will call this `$RESTORE_DIR` in the instructions below.

```
sudo cp ~/$RESTORE_DIR/securedrop-workstation-dom0-config/{sd-journalist.sec,
↪config.json,sd-keys.asc} /usr/share/securedrop-workstation-dom0-config/
```

Optionally, inspect each file before proceeding. The first file should be an ASCII-armored GPG private key file. The second file should follow the format of the example configuration file, with values for its fields (e.g., `hostname`, `submission_key_fpr`) specific to your configuration. The file may be formatted in a single line without whitespace. The third file is a backup of key material from `sd-gpg` and will be moved into that VM when you have reprovisioned the system.

Verify that the configuration is valid:

```
sdw-admin --validate
```

If the above command prints `OK`, the configuration is valid.

Reinstall SecureDrop Workstation:

```
sdw-admin --apply
```

Restore additional keys to `sd-gpg`

In a `dom0` terminal:

```
qvm-copy-to-vm sd-gpg $RESTORE_DIR/securedrop-workstation-dom0-config/sd-keys.
↪asc
qvm-run sd-gpg 'gpg --import /home/user/QubesIncoming/dom0/sd-keys.asc'
```

Restore customized VMs, RPC policies

At this stage, you should have a functional SecureDrop Workstation. You may restore any additional customizations or additional VMs, being mindful that you are responsible for the security implications of customizing this system.

Customizations in `dom0` must be restored manually, meaning that any RPC policies you have added will need to be moved into place from the `$RESTORE_DIR`.

Once you are finished with the `$RESTORE_DIR` and have verified that your system works (download, decrypt, sync), you may delete the `$RESTORE_DIR`.

(Post-migration instructions) Destroy backup medium

Wipe (reformat) the LUKS-encrypted storage device that you used to store SecureDrop Workstation configuration material, overwriting the LUKS header and all data with a new encrypted partition, or physically destroy the backup medium, to ensure you are not proliferating copies of sensitive data.

1.66 BIOS update instructions

1.66.1 Automatic BIOS updates

These instructions should work for many recent laptops, including the two ThinkPad models specifically included in our *Hardware*.

If your laptop has Ubuntu preinstalled, run its **Software Updater** twice as follows:

1. to install software updates, especially for the `fwupd` package; and then
2. to run `fwupd` to update the BIOS automatically.

If **Software Updater** offers to run `fwupd` during step (1), decline until step (2), to make sure `fwupd` itself has received its latest security updates.

Other Linux

If your laptop has another Linux distribution installed, use the built-in software manager (such as GNOME Software or KDE Discover) to update the available software. Most modern distributions include `fwupd` by default. If not, you can install the package using your preferred software manager.

Once `fwupd` is installed, you can install available updates by running:

```
fwupdmgr refresh
fwupdmgr update
```

1.66.2 Manual BIOS updates

If your laptop is not supported by `fwupd`, you will need to consult the manual for your specific make and model to determine how to manually apply a BIOS update. The process will likely include downloading an update file, verifying its integrity, copying it to a USB flash drive, and then accessing an update menu within the BIOS settings. If you have a Thinkpad, refer to the instructions for *Manual BIOS on Lenovo ThinkPad laptops*.

Manual BIOS on Lenovo ThinkPad laptops

The instructions below assume the use of a Linux-based computer for the creation of a BIOS upgrade USB flash drive. To upgrade the BIOS:

- Locate the ThinkPad's "machine type" in its BIOS setup program:
  1. Boot (or reboot) the ThinkPad and follow the prompts to enter setup, usually via the <Enter> and <F1> keys.
  2. On the **Main** tab, look for the **Machine Type Model**. The first four characters, such as *20L5*, *20L6*, or *20S0*, are the machine type.
- Visit https://support.lenovo.com in the Linux-based computer. Type the machine type found above into the search bar, then press **Enter**.
- In the "Product Home" page, select **Drivers And Software** and choose **BIOS/UEFI**.
- Download the file called either **BIOS Update (Bootable CD)** or **BIOS Update (Utility & Bootable CD)**.

Note

A Tails drive can be used for the verification and conversion process described below, but the Lenovo support site blocks requests over Tor, preventing the ISO download. To work around this, either:

- download the BIOS ISO on a different computer and transfer it to Tails using another USB flash drive, or
- download the ISO in Tails using the Unsafe Browser as follows:
  - Start Tails with an administration password set and the Unsafe Browser enabled under "Additional Settings" on the Welcome Screen.
  - Open the Unsafe Browser: **Applications ▶ Internet ▶ Unsafe Browser** and find and download the ISO

  - Note the filename, as you'll need it for subsequent steps.
  - Leave the Unsafe Browser running, and open a terminal via **Applications ▶ System Tools ▶ Terminal**.
  - Copy the ISO to the desktop with the command:

```
sudo cp /var/lib/unsafe-browser/chroot/home/clearnet/Downloads/<fileName.iso>
↪ ~amnesia/Desktop
```

  - Fix the ISO file's ownership with the command:

```
sudo chown amnesia:amnesia ~amnesia/Desktop/<fileName.iso>
```

- Verify the checksum of the downloaded ISO file using the following command, comparing it against the checksum in the file listing above:

```
sha256sum /path/to/downloaded.iso
```

- Create a USB-bootable version of the ISO using the command:

```
geteltorito <path/to/CDISO> > usb-bios.iso
```

Note

To install the `geleltorito` utility on Debian-based systems, use the command

```
sudo apt install genisoimage
```

To install it on Fedora-based systems, use the command:

```
sudo dnf install geteltorito genisoimage
```

- Plug in a USB flash drive and check its device name with the `lsblk` command - use the root device name below, not a partition (eg. `/dev/sdc` instead of `/dev/sdc1`).
- Write the BIOS update ISO to the USB flash drive using the following command:

```
sudo dd if=usb-bios.iso of=/dev/sdX bs=1M && sync
```

where `sdX` is the device name verified above.

Caution

The `dd` command will wipe data on the targeted device. Make sure that you use the correct device name.

Once complete, remove the USB flash drive.

- Plug the USB flash drive into the ThinkPad.
- Boot the ThinkPad and follow the prompts to enter its startup and boot menus, likely via the <Enter> and <F12> keys, respectively.
- Follow the on-screen instructions to update the BIOS, including any mandatory reboots. Note that the instructions may refer to an update CD instead of your update USB flash drive.

1.67 Reviewing and exporting logs

The *Journalist Workstation* aggregates system logs from all its VMs in the `sd-log` VM, in the folder `~/QubesIncomingLogs`, with one subfolder for each VM. You can inspect these logs directly in the `sd-log` VM, or you can copy them to another VM, e.g., for purposes of sharing logs with the SecureDrop development team.

Please note that while the logs do not include original filenames or message contents, they do contain sensitive information, e.g.:

- timing and usage information related to SecureDrop access
- the two-word designation for a given *Source*
- metadata about submissions and replies
- error messages that disclose further details

For this reason, the `sd-log` VM is networkless, and you cannot copy files from `sd-log` to other VMs by default.

If you want to selectively enable copying logs to a single VM, you can use tags, similar to the method used for *managing clipboard access*. You can add and remove the permission just before each copying operation; the change will take effect immediately.

Important

Before copying logs to a networked VM, inspect them for sensitive information, and redact them as warranted.

To enable copying logs to a target VM, you can use a command like the following in `dom0`, substituting `<VM name>` with the name of the target VM (e.g., `work`):

```
qvm-tags <VM name> add sd-receive-logs
```

Verify that the tag was successfully applied using the `ls` subcommand:

```
qvm-tags <VM name> ls
```

To remove the permission, use this command in `dom0`:

```
qvm-tags <VM name> del sd-receive-logs
```

With the permission in effect, you can use the command `qvm-copy` in a terminal in `sd-log` to copy individual files to the target VM. For example, to copy a file `syslog-redacted.log`, you would use this command:

```
qvm-copy syslog-redacted.log
```

A graphical prompt will permit you to select any target VM that has the `sd-receive-logs` tag. Once successfully copied, the file can be found in the directory `~/QubesIncoming/sd-log` in the target VM. See the Qubes OS documentation on copying files for more information.

To review current copy permissions, you can use `qvm-ls` to print out a list of VMs that can receive files from `sd-log`:

```
qvm-ls --tags sd-receive-logs
```

1.68 Troubleshooting system updates

After you log into Qubes, the preflight updater will prompt you to check for available system updates at least once per day.

If updates fail for any reason, the preflight updater will not launch SecureDrop Inbox until the underlying issue has been resolved. This is to ensure that the system is in a secure state before you interact with SecureDrop.

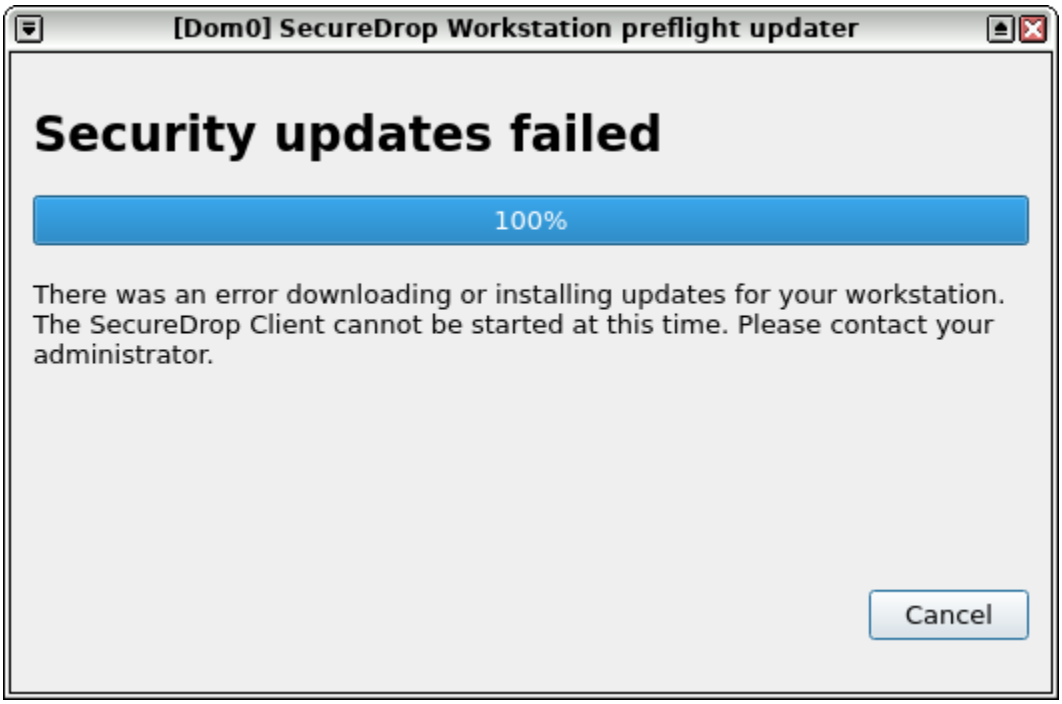


Fig. 4: The error displayed when the preflight updater does not successfully complete the update.

This guide offers troubleshooting steps for common update issues.

1.68.1 Step 1: Locate the updater log

The preflight updater runs in the `dom0` domain. It writes its log to `~/.securedrop_updater/logs/updater.log`. Log files are rotated hourly; if you have started the updater again since the error occurred, you may need to check the previous log file.

In order to examine the most recent log file:

1. Open a terminal in `dom0` via  ▶ **Gear Icon (left-hand side)** ▶ **Other Tools** ▶ **Xfce Terminal**.
2. Change to the `~/.securedrop_updater/logs/` directory:

   ```
   cd ~/.securedrop_updater/logs/
   ```

3. Display the most recent log file:

   ```
   cat updater.log
   ```

In order to locate a previous log file in the same directory:

1. Locate the most recently modified log file.

   ```
   ls -t updater.log* | head -n 2
   ```

2. Display the file that ends with a date and time stamp, e.g.:

   ```
   cat updater.log.2023-01-01_10
   ```

1.68.2 Step 2: Identify the cause(s) of the error

If the updater has run to completion, you should see a result line in the log file that looks similar to the following:

```
2025-02-24 20:12:11,821 - sd.sdw_updater_gui.UpdaterApp:71(upgrade_status)
INFO: Signal: upgrade_status {
'dom0': <UpdateStatus.UPDATES_OK: '0'>,
'apply_dom0': <UpdateStatus.UPDATES_OK: '0'>,
'fedora-42-xfce': <UpdateStatus.UPDATES_FAILED: '3'>,
'sd-large-bookworm-template': <UpdateStatus.UPDATES_OK: '0'>,
'sd-small-bookworm-template': <UpdateStatus.UPDATES_OK: '0'>,
'recommended_action': <UpdateStatus.UPDATES_FAILED: '3'>}
```

In this example, the `fedora-42-xfce` VM has failed to update. This is indicated by the text `<UpdateStatus.UPDATES_FAILED: '3'>`.

It is possible that multiple steps have failed. Make note of any of the individual steps that have failed, other than `recommended_action`.

1.68.3 Step 3: Resolve the issue(s)

The resolution path will depend on which step(s) failed. Note that `dom0` and `apply_dom0` are separate steps.

`dom0` update failures

1. Open a terminal in `dom0` via  ► **Gear Icon (left-hand side)** ► **Other Tools** ► **Xfce Terminal**.
2. Perform an interactive `dom0` update by running the following command:

   ```
   sudo qubes-dom0-update
   ```

3. Follow the prompts to resolve any issues. If you are unsure on how to resolve an error, please contact us for assistance.
4. Reboot the system. `dom0` updates are often security-sensitive, and may require a reboot to take effect.

Expired SecureDrop signing key

If the update fails after running `sudo qubes-dom0-update` as described above, and the terminal console displays the following message:

```
1. Certificiate 188EDD3B7B22E6A3 invalid: certificate is not alive
    because: The primary key is not live
    because: Expired on 2023-07-04T10:52:20Z
2. Key 188EDD3B7B22E6A3 invalid: key is not alive
    because: The primary key is not live
    because: Expired on 2023-07-04T10:52:20Z
[...]
Error: GPG check FAILED
```

your system is trying to use an old copy of the SecureDrop Release Signing Key. You can perform the following steps to fetch the updated key and remove the expired one:

1. **Start a terminal** in the "work" VM via the menu:  ▶ **Apps** ▶ **work** ▶ **Xfce Terminal**
2. **Download the key:**

   Run command:

   ```
   gpg --keyserver hkps://keys.openpgp.org --recv-key "2359 E653 8C06 13E6 5295 5E6C␣
   ↪188E DD3B 7B22 E6A3"
   ```

   Expected output:

   ```
   gpg: key 188EDD3B7B22E6A3: public key "SecureDrop Release Signing Key <securedrop-
   ↪release-key-2021@freedom.press>" imported
   gpg: Total number processed: 1
   gpg: imported: 1
   ```

3. **Verify the expiry is 2027-05-24:**

   Run command:

   ```
   gpg -k securedrop
   ```

   Expected output:

   ```
   pub   rsa4096 2021-05-10 [SC] [expires: 2027-05-24]
      2359E6538C0613E652955E6C188EDD3B7B22E6A3
   uid           [ unknown] SecureDrop Release Signing Key <securedrop-release-key-
   ↪2021@freedom.press>
   sub   rsa4096 2021-05-10 [E] [expires: 2027-05-24]
   ```

4. **Export the downloaded key:**

   Run command:

   ```
   gpg --armor --export "2359 E653 8C06 13E6 5295 5E6C 188E DD3B 7B22 E6A3" >␣
   ↪securedrop-release-key.pub
   ```

   No output expected.

5. **Print the exported key's checksum:**

   Run command:

   ```
   sha256sum securedrop-release-key.pub
   ```

   Expected output:

   ```
   fedef93de425668541545373952b5f92bac4ac1f1253fe5b64c2be2fc941073b securedrop-release-
   ↪key.pub
   ```

6. **Start a dom0 terminal** via  ▶  ▶ **Other Tools** ▶ **Xfce Terminal**.

   The remaining commands will all be executed in this dom0 terminal.

7. **Copy the key into dom0:**

   Run command:

```
qvm-run --pass-io work cat securedrop-release-key.pub > /tmp/securedrop-release-key.
↪pub
```

No output expected.

8. **Verify the key checksum matches:**

   Run command:

```
sha256sum /tmp/securedrop-release-key.pub
```

   Expected output:

```
fedef93de42566854154537395 2b5f92bac4ac1f1253fe5b64c2be2fc941073b /tmp/securedrop-
↪release-key.pub
```

9. **Copy the key into place:**

   Run command:

```
sudo cp /tmp/securedrop-release-key.pub /etc/pki/rpm-gpg/RPM-GPG-KEY-securedrop-
↪workstation
```

   No output expected.

10. **Delete the old key from RPM:**

    Run command:

```
sudo rpm -e gpg-pubkey-7b22e6a3-609966ad
```

    No output expected.

11. **Import the new key into RPM:**

    Run command:

```
sudo rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY-securedrop-workstation
```

    No output expected.

12. **Verify the expiry is 2027-05-24:**

    Run command:

```
gpg --show-keys /etc/pki/rpm-gpg/RPM-GPG-KEY-securedrop-workstation
```

    Expected output:

```
pub   rsa4096 2021-05-10 [SC] [expires: 2027-05-24]
   2359E6538C0613E652955E6C188EDD3B7B22E6A3
uid           [ unknown] SecureDrop Release Signing Key <securedrop-release-
↪key-2021@freedom.press>
sub   rsa4096 2021-05-10 [E] [expires: 2027-05-24]
```

`sd-*-template` update failures

1. Click the Qubes menu and open a terminal in the impacted template. For example, if `sd-small-bookworm-template` failed to update, select its entry in the Qubes menu and click **Terminal**.
2. Perform an interactive template update by running the the following commands:

   ```
   sudo apt update
   ```

   ```
   sudo apt upgrade
   ```

   The SecureDrop and Whonix templates are based on Debian GNU/Linux. The `apt update` comand will ensure the package index is up-to-date, and the `apt upgrade` comand will apply updates.
3. Follow the prompts to resolve any issues. If you are unsure on how to resolve an error, please contact us for assistance.

`fedora-42-xfce` update failures

1. Launch the Qubes GUI Updater from the top righthand tray icon. Ensure the `fedora-42-xfce` template is selected.
2. Run the updater, observing the output in the updater dialog.
3. If the update is not successful, contact Support and provide the output you see in the dialog.

`apply_dom0` update failures

The `apply_dom0` step applies any necessary configuration changes to the SecureDrop Workstation. If this step fails, this may indicate a misconfiguration, or it could be a result of download failures during the operation.

We recommend first re-running the updater by double-clicking the SecureDrop desktop icon. This may resolve transient network issues.

If this does not resolve the issue:

1. Locate the `updater-detail.log` file in the same directory as the `updater.log` file. This file contains more detailed information about the `apply_dom0` step.

   Like the `updater.log` file, this file is rotated hourly.
2. Copy this file to a networked VM by using the `qvm-copy-to-vm` command. For example, to copy the file to the `work` VM:

   ```
   qvm-copy-to-vm work ~/.securedrop_updater/logs/updater-detail.log
   ```
3. The file can now be found in `~/QubesIncoming/dom0/` in the `work` VM.

   Send us the file through a secure channel, such as via Signal. We will provide further instructions.

1.68.4 Step 4: Restart the updater

Click the SecureDrop Inbox desktop icon to restart the updater. If all issues have been resolved, the updater should run to completion and display a success message. If the issue persists, please contact us for assistance.

1.69 Managing clipboard access

Every VM in Qubes has its own clipboard, similar to the clipboard of a Mac, Windows or Linux computer. For example, if you used the default `work` VM to browse the web and wanted to copy text from one browser window to another, you would use the `Ctrl+C` and `Ctrl+V` keyboard shortcuts to copy and paste. This type of clipboard usage – copy and paste in the same VM – also works in all VMs that are part of a SecureDrop Workstation.

In addition, Qubes supports copying information *between* VMs. This is done by using special keyboard shortcuts, `Ctrl+Shift+C` and `Ctrl+Shift+V`, in a four-step process. By default, this is disabled for all VMs that are part of a SecureDrop Workstation, consistent with the principle of least privilege.

As an administrator, you should be aware of the following risks related to clipboard access before changing the default configuration:

1. It is dangerous to copy untrusted, unsanitized content *into* a secure environment. What looks like plain text may contain character sequences that exploit security vulnerabilities in the target environment.
2. The four-step process described above can be difficult to follow, and it is easy to make an operational mistake, such as pasting a password into a message to a *Source*, or into a window belonging to a VM with network access.
3. Like any other part of the operating system, the implementation of Qubes clipboard itself may contain undiscovered security vulnerabilities that an adversary could exploit in an attempt to exfiltrate information.

With these considerations in mind, there are use cases where clipboard access may be an important part of your regular use of SecureDrop Workstation. For example:

- You may want to copy passwords from a password manager to log into SecureDrop Inbox;
- You may want to copy a message you received via SecureDrop into a secure messaging app like Signal, to share it with another *Journalist*.

To support these use cases, Qubes OS allows you to grant granular access to the `sd-app` clipboard (via the cross-VM clipboard) to selected VMs.

1.69.1 Configuring clipboard access to `sd-app`

The process for permitting the one-directional copying of passwords from a password manager in `vault` to SecureDrop Inbox is *outlined in the installation docs*. In general, clipboard access to SecureDrop Workstation VMs is governed by *tags* that can be applied in `dom0` to selected VMs:

- the tag `sd-send-app-clipboard` can be used to tag a VM that should be able to send its clipboard contents *to* `sd-app` via the cross-VM clipboard;
- the tag `sd-receive-app-clipboard` can be used to tag a VM that should be able to receive its clipboard contents *from* `sd-app` via the cross-VM clipboard.

You can configure these tags for a given VM from the `dom0` terminal. Changes to tags take effect immediately, and any VM can have multiple tags.

Important

Make sure you fully understand technical and operational security risks before permitting clipboard access to any VM. The "send" and "receive" tags are separate so you can set up only the clipboard direction you need to support a given use case.

We recommend adding a note about any changes to the clipboard configuration to your internal documentation for SecureDrop. If you are unsure how to configure the clipboard to support a specific use case, please do not hesitate to contact us for assistance.

The general syntax for adding a tag is as follows, substituting `<VM name>` with the name of an existing VM in the system you want to grant access to the clipboard:

```
qvm-tags <VM name> add <tag name>
```

Confirm that the command was successfully applied using the `ls` subcommand:

```
qvm-tags <VM name> ls
```

The syntax for revoking a tag is as follows:

```
qvm-tags <VM name> del <tag name>
```

As before, confirm the operation via the `ls` subcommand.

As an example, if you had a custom VM called `work-signal` that runs the Signal messenger, and you wanted to copy and paste messages from SecureDrop Inbox *into* Signal (and potentially other applications in that VM) but not *out* of Signal into SecureDrop Inbox, you would issue the following commands:

```
qvm-tags work-signal add sd-receive-app-clipboard
qvm-tags work-signal ls
```

To review current clipboard permissions, you can use `qvm-ls` to print out a list of VMs that can receive or send clipboard contents:

```
qvm-ls --tags sd-receive-app-clipboard
qvm-ls --tags sd-send-app-clipboard
```

1.70 Glossary

A number of terms used in this guide, and in the *SecureDrop workflow diagram* <*what_is_securedrop*>, are specific to SecureDrop. The list below attempts to enumerate and define these terms.

1.70.1 Admin Workstation

1.70.2 Application Server

The *Application Server* runs the SecureDrop server application. This server hosts both the website that *Sources* access (the *Source Interface*) and the website that *Journalists* access (the *Journalist Interface*). Both are published through an *Onion Service* because *Sources*, *Journalists*, and admins may only connect to this server using Tor.

1.70.3 Export Device

The *Export Device* is the physical media (e.g., designated USB flash drive) used to transfer decrypted documents from the *Secure Viewing Station* to a *Journalist*'s everyday workstation, or to another computer for additional processing.

Please see the detailed security recommendations for the choice, configuration and use of your *Export Device* in the *journalist* guide and in the *setup guide*.

1.70.4 Journalist

The *Journalist* uses SecureDrop to communicate with and download documents submitted by the *Source*. *Journalists* do this by using the *SecureDrop Workstation*.

If a *Journalist* chooses to release any of these documents, they can be prepared for publication on the *SecureDrop Workstation* before being transferred to an Internet-connected computer.

Instructions for using SecureDrop as a *Journalist* are available in our *Journalist Guide*.

1.70.5 Journalist Alert Public Key

The *Journalist Alert Public Key* is used for encrypting the daily alert that notifies *Journalists* via encrypted email about whether or not there has been submission activity in the past 24 hours. The *Journalist* uses an associated private key to decrypt the alerts.

1.70.6 Journalist Workstation

1.70.7 Landing Page

The *Landing Page* is the public-facing webpage for a SecureDrop instance. This page is hosted as a standard (i.e. non-Tor) webpage on the news organization's site. It provides first instructions for potential *Sources* and includes the instance's *Source Interface* address.

1.70.8 Monitor Server

The *Monitor Server* keeps track of the *Application Server* and sends out an email alert if something seems wrong. Only system admins connect to this server, and they may only do so using Tor.

1.70.9 Onion Service

Tor *Onion Services* provide anonymous inbound connections to websites and other servers exclusively over the Tor network. For example, SecureDrop uses onion services for the *Journalist Interface* and *Source Interface* websites, as well as for administrative access to the servers in SSH-over-Tor mode.

Onion Services can be accessed by clicking a link or pasting the *Onion Service* address into Tor Browser. For example, `sdolvtfhatvsysc6l34d65ymdwxcujausv7k5jk4cy5ttzhjoi6fzvyd.onion` is the onion service address for the SecureDrop website.

Read more about *Onion Services* in Tor's glossary.

1.70.10 OSSEC Alert Public Key

The *OSSEC Alert Public Key* is the GPG key that OSSEC will encrypt alerts to. The associated private key is used by the admin to access encrypted OSSEC alerts from the *Monitor Server*.

1.70.11 Source

The *Source* is the person who submits documents to SecureDrop and may use SecureDrop to communicate with a *Journalist*. A *Source* will always access SecureDrop through the *Source Interface* and must do so using Tor.

Instructions for using SecureDrop as a *Source* are available in our *Source Guide*.

1.70.12 Source Interface

The *Source Interface* is the website that *Sources* will access to submit documents and communicate with *Journalists*. This site is hosted on the *Application Server* and can only be accessed through Tor.

Instructions for using the *Source Interface* are available in our *Source Guide*.

1.70.13 Submission Key

The *Submission Key* is the GPG keypair used to encrypt and decrypt documents and messages sent to your SecureDrop. Because the public key and private key must be treated very differently, we sometimes refer to them explicitly as the *Submission Public Key* and the *Submission Private Key*.

The *Submission Public Key* is uploaded to your SecureDrop servers as part of the installation process. Once your SecureDrop is online, anyone will be able to download it.

The *Submission Private Key* should never be accessible to a computer with Internet connectivity. Instead, it should remain on the *Secure Viewing Station* and on offline backup storage.

1.70.14 Two-Factor Authentication

There are several places in the SecureDrop architecture where two-factor authentication is used to protect access to sensitive information or systems. These instances use the standard TOTP and/or HOTP algorithms, and so a variety of devices can be used to generate 6-digit two-factor authentication codes. We recommend using one of:

- FreeOTP for Android or for iOS installed
- A YubiKey

Tip

We recommend using FreeOTP (available for Android and for iOS) to generate two-factor codes because it is Free Software. However, if it does not work for you for any reason, alternatives exist:

- Google Authenticator for Android and iOS (proprietary)
- authenticator for the desktop (Free Software)

1.71 Threat model

This document outlines the threat model for SecureDrop 0.3 and is inspired by a document Adam Langley wrote for Pond. The threat model is defined in terms of what each possible adversary can achieve. This document is always a work in progress. If you have questions or comments, please open an issue on GitHub or send an email to securedrop@freedom.press.

The threat model for the SecureDrop Workstation based on Qubes OS is summarized in a separate document.

1.71.1 Actors

The SecureDrop ecosystem comprises a host of actors, organized by the following high-level categories: *Users*, *Adversaries*, and *Systems*.

Users

The following table of the users who interact with the SecureDrop web application. Note that the airgapped *Secure Viewing Station* with the GPG *Submission Key* is required to decrypt submissions or messages.

User Type	Trust Level
Source	• Submit a document or message
Recurring source	• Submit another document or message • Read replies
Journalist	• Download *all* GPG-encrypted documents from *all* sources • Download *all* GPG-encrypted messages from *all* sources • Reply to *all* sources
Admin	• Download *all* GPG-encrypted documents from *all* sources • Download *all* GPG-encrypted messages from *all* sources • Reply to *all* sources • Change the SecureDrop instance logo • SSH and root privileges on *app* and *mon* servers

Adversaries

We consider the following classes of attackers for the design and assessment of SecureDrop:

Adversary	Capabilities
Nation State / Law Enforcement / Global Adversary	• Large scale, full-packet network capture • Active network attacks • Advanced attacks on infrastructure • Hardware and software implants for persistence • Cryptanalysis • Exploitation of unknown vulnerabilities
Large Corporation	• Limited network capture • Some targeted attacks on infrastructure • Use of known vulnerabilities • Mostly limited to software-based attacks
Internet Service Provider	• Full network capture • Mostly limited to network-based attacks
User Error	• Source, Journalist, Administrator or Developer error
Dedicated Individual	• Use of known vulnerabilities • Mostly limited to software-based attacks

Systems

For more information about the various systems involved in a SecureDrop deployment, please visit the *hardware section*.

System	Description
Hardware Firewall	• Dedicated Hardware Firewall • pfSense-based • 3 Interfaces: *app*, *mon* and *admin*
Application Server	• SecureDrop Source Interface • SecureDrop Journalist Interface • SSH Server • Ossec Client
Monitor Server	• Ossec Server • SSH Server
Journalist/Admin Workstation	• Internet-connected laptop • Tails USB with persistence volume
Secure Viewing Station	• Airgapped and stripped-down laptop • Tails USB with persistence volume

1.71.2 Assumptions

The following assumptions are accepted in the threat model of every SecureDrop project:

Assumptions about the *Source*

- The *Source* acts reasonably and in good faith, e.g. if the *Source* were to give their credentials or private key material to the attacker that would be unreasonable.
- The *Source* would like to remain anonymous, even against a forensic attacker.
- The *Source* obtains an authentic copy of Tails and Tor Browser.
- The *Source* follows our *guidelines* for using SecureDrop.

- The *Source* is accessing an authentic SecureDrop site.

Assumptions about the admin and the *Journalist*

- The admin and the *Journalist* act reasonably and in good faith, e.g. if either of them were to give their credentials or private key material to the attacker that would be unreasonable.
- The admin and the *Journalist* obtain authentic copies of Tails.
- The *Journalist* follows our *guidelines* for using SecureDrop and working with submitted documents.

Assumptions about the person installing SecureDrop

- This person (usually the admin) acts reasonably and in good faith, e.g. if they were to give the attacker system-level access that would be unreasonable.
- The person obtains an authentic copy of SecureDrop and its dependencies.
- The person follows our guidelines for *deploying the system*, setting up the *landing page* for the organization, and for *installing SecureDrop*.

Assumptions about the *Source*'s computer

- The computer correctly executes Tails or Tor Browser.
- The computer is not compromised by malware.

Assumptions about the *Admin Workstation* and the *Journalist Workstation*

- The computer correctly executes Tails.
- The computer and the Tails device are not compromised by malware.
- The *Two-Factor Authentication* device used with the workstation are not compromised by malware.

Assumptions about the *Secure Viewing Station*

- The computer is airgapped.Onion
- The computer correctly executes Tails.
- The computer and the Tails device are not compromised by malware.

Assumptions about the SecureDrop hardware

- The servers correctly execute Ubuntu, SecureDrop and its dependencies.
- The servers, network firewall, and physical media are not compromised by malware.

Assumptions about the organization hosting SecureDrop

- The organization wants to preserve the anonymity of its *Sources*.
- The organization acts in the interest of allowing *Sources* to submit documents, regardless of the contents of these documents.
- The users of the system, and those with physical access to the servers, can be trusted to uphold the previous assumptions unless the entire organization has been compromised.
- The organization is prepared to push back on any and all requests to compromise the integrity of the system and its users, including requests to deanonymize *Sources*, block document submissions, or hand over encrypted or decrypted submissions.

Assumptions about the world

- The security assumptions of RSA (4096-bit GPG and SSH keys) are valid.
- The security assumptions of scrypt with randomly-generated salts are valid.
- The security/anonymity assumptions of Tor and the *Onion Service* protocol are valid.
- The security assumptions of the Tails operating system are valid.
- The security assumptions of SecureDrop dependencies, specifically Ubuntu, the Linux kernel, application packages, application dependencies are valid.

Other assumptions or factors

- The level of press freedom may vary in both geography and time.
- The number of daily Tor users in a country can greatly vary.

1.71.3 Assets

Asset Type	Asset
Assets relating to SecureDrop users	• Login details • Encryption key(s) • SSH details
Assets relating to the publicly accessed system	• Access to documents via server • Access to documents via Journalist Interface • Access to admin privileges via Journalist Interface • Access to user alerts, support tickets
Assets relating to the underlying system	• SecureDrop code manipulation • Dependency code manipulation

1.71.4 Implications of SecureDrop area compromise

What a compromise of the *Application Server* can surrender

- The server sees the plaintext codename, used as the login identifier, of every *Source*.
- The server sees all HTTP requests made by the*Source*, the admin, and the *Journalist*.
- The server sees the plaintext submissions of every *Source*.
- The server sees the plaintext communication between *Journalists* and their *Sources*.
- The server stores the onion service private key for the *Source* interface.
- The server stores the onion service private key and authentication token for the *Journalist Interface*.
- The server stores and (optional) TLS private key and certificate (if HTTPS is enabled on the *Source* interface)
- The server stores hashes of codenames, created with scrypt and randomly-generated salts.
- The server stores journalist password hashes, created with script and randomly-generated salts, as well as TOTP seeds.
- The server stores only encrypted submissions and communication on disk.
- The server stores a GPG key for each *Source*, with the *Source*'s codename as the passphrase.
- The server may store plaintext submissions in memory for at most 24 hours.

- The server stores sanitized Tor logs, created using the SafeLogging option, for the *Source Interface*, the *Journalist Interface*, and SSH.
- The server stores both access and error logs for the *Journalist Interface*.
- The server stores connection history and audit logs for the admin.
- The server can connect to the *Monitor Server* using an SSH key and a passphrase.

What a compromise of the *Monitor Server* can surrender

- The server stores the plaintext alerts on disk, data may also reside in RAM.
- The server stores the *OSSEC Alert Public Key* the OSSEC alerts are encrypted to.
- The server stores plaintext credentials for the SMTP relay used to send OSSEC alerts.
- The server stores the email address the encrypted OSSEC alerts are sent to.
- The server stores sanitized Tor logs, created using the SafeLogging option, for SSH.
- The server stores connection history and audit logs for the admin.
- The server stores OSSEC and Procmail logs on disk.
- The server can connect to the *Application Server* using an SSH key and a passphrase.

What a compromise of the workstations can surrender

- The *Admin Workstation* requires Tails with a persistent volume, which stores information such as GPG and SSH keys, as well as a *database with passphrases* for the *Application Server*, the *Monitor Server*, and the GPG key the *Monitor Server* will encrypt OSSEC alerts to.
- The *Journalist Workstation* requires Tails with a persistent volume, which stores information such as the onion service value required to connect to the *Journalist Interface*, as well as a *database with passphrases* for the *Journalist Interface*.
- The *Secure Viewing Station* requires Tails with a persistent volume, which stores information such as the SecureDrop application's GPG key, as well as a *database with the passphrase* for that key.

What a compromise of the *Source*'s property can surrender

- Use of Tor Browser will leave traces that can be discovered through a forensic analysis of the *Source*'s property following either a compromise or physical seizure. Unless the compromise or seizure happens while the *Source* is submitting documents to SecureDrop, the traces will not include information about sites visited or actions performed in the browser.
- Use of Tails with a persistent volume will leave traces on the device the operating system was installed on. Unless the compromise or seizure happens while the *Source* is submitting documents to SecureDrop, or using the persistent volume, the traces will not include information about sites visited or actions performed in the browser or on the system.
- SecureDrop 0.3 encourages *Sources* to protect their codenames by memorizing them. If a *Source* cannot memorize the codename right away, we recommend writing it down and keeping it in a safe place at first, and gradually working to memorize it over time. Once the *Source* has memorized it, they should destroy the written copy. If the *Source* does write down the codename, a compromise or physical seizure of the *Source*'s property may result in the attacker obtaining the *Source*'s codename.
- An attacker with access to the **source's codename** can:
  - Show that the *Source* has visited the SecureDrop site, but not necessarily submitted anything.
  - Upload new documents or submit messages.

  - Communicate with the *Journalist* as that *Source*.
  - See any replies from *Journalists* that the *Source* has not yet deleted.

What a physical seizure of the *Source*'s property can surrender

- Document use of Tor or Tails, but not necessarily research into SecureDrop
- Prevent the *Source* from submitting documents by taking the device the documents are stored on.
- If the property is seized while powered on, the attacker can also analyze any plaintext information that resides in RAM.
- Tamper with the hardware.
- A physical seizure of, and access to, the *Source*'s codename will allow the attacker to access the *Source Interface* as that *Source*.
- A physical seizure of the admin's property will allow the attacker to:
  - Prevent the admin from working on SecureDrop for some period of time.
  - Access any stored, decrypted documents taken off the Secure Viewing Station.
  - If the property is seized while powered on, the attacker can also analyze any plaintext information that resides in RAM.
- A physical seizure of, and access to, the admin's Tails persistent volume, password database, and *Two-Factor Authentication* device will allow the attacker to access both servers and the *Journalist Interface*.

What compromise of the admin's property can surrender

- To access the *Journalist Interface*, the *Application Server*, or the *Monitor Server*, the attacker needs to obtain the admin's login credentials and the admin's *Two-Factor Authentication* device. Unless the attacker has physical access to the servers, the attacker will also need to obtain the onion service values for the Interface and the servers. This information is stored in a password-protected database in a persistent volume on the admin's Tails device. The volume is protected by a passphrase. If the admin's two-factor authentication device is a mobile phone, this will also be protected by a passphrase.
- An attacker with access to the **admin's computer** can:
  - Access any stored, decrypted documents taken off the Secure Viewing Station.
- An attacker with access to the **persistent volume** on the admin's Tails device can:
  - Add, modify, and delete files on the volume.
  - Access the onion service values used by the Interfaces and the servers.
  - Access SSH keys and passphrases for the *Application Server* and the *Monitor Server*.
  - Access the GPG key and passphrase for the encrypted OSSEC email alerts.
  - Access the credentials for the account the encrypt alerts are sent to.
  - Access the admin's personal GPG public key, if stored there.
- An attacker with admin access to the *Journalist Interface* can:
  - Add, modify, and delete journalist users.
  - Change the codenames associated with *Sources* within the Interface.
  - Download, but not decrypt, submissions.
  - Communicate with *Sources*.

  - Delete one or more submissions.
  - Delete one or more *Sources*, which destroys all communication with that *Source* and prevents the *Source* from ever logging back in with that codename.
- An attacker with admin access to the *Application Server* can:
  - Add, modify, and delete software, configurations, and other files.
  - See all HTTP requests made by the *Source*, the admin, and the *Journalist*.
  - See the plaintext codename of a *Source* as they are logging in.
  - See the plaintext communication between a *Source* and a *Journalist* as it happens.
  - See the stored list of hashed codenames.
  - Access the GPG public key used to encrypt communications between a *Journalist* and a *Source*.
  - Download stored, encrypted submissions and replies from the *Journalists*.
  - Decrypt replies from the *Journalists* if the *Source*'s codename, and thus the passphrase, is known.
  - Analyze any plaintext information that resides in RAM, which may include plaintext of submissions made within the past 24 hours.
  - Review logs stored on the system.
  - Access the *Monitor Server*.
- An attacker with admin access to the *Monitor Server* can:
  - Add, modify, and delete software, configurations, and other files.
  - Change the SMTP relay, email address, and GPG key used for OSSEC alerts.
  - Analyze any plaintext information that resides in RAM.
  - Review logs stored on the system.
  - Trigger arbitrary commands to be executed by the OSSEC agent user, which, assuming the attacker is able to escalate privileges, may affect the *Application Server*.

What a physical seizure of the admin's property can achieve

- Tamper with the hardware.
- Prevent the admin from working on SecureDrop for some period of time.
- Access any stored, decrypted documents taken off the Secure Viewing Station.
- If the property is seized while powered on, the attacker can also analyze any plaintext information that resides in RAM.
- A physical seizure of, and access to, the admin's Tails persistent volume, password database, and *Two-Factor Authentication* device will allow the attacker to access both servers and the *Journalist Interface*.

What a compromise of the *Journalist*'s property can achieve

- To access the *Journalist Interface*, the attacker needs to obtain the *Journalist*'s login credentials and the *Journalist*'s *Two-Factor Authentication* device or seed. Unless the attacker has physical access to the server, the attacker will also need to obtain the onion service value for the *Interface*. This information is stored in a password-protected database in a persistent volume on the *Journalist*'s Tails device. The volume is protected by a passphrase. If the *Journalist*'s *Two-Factor Authentication* device is a mobile phone, this will also be protected by a passphrase.
- An attacker with access to the *Journalist*'s computer can:

  - Access any stored, decrypted documents taken off the Secure Viewing Station.
- An attacker with access to the **persistent volume** on the *Journalist*'s Tails device can:
  - Add, modify, and delete files on the volume.
  - Access the onion service values used by the *Journalist Interface*.
  - Access SSH keys and passphrases for the *Application Server* and the *Monitor Server*.
- An attacker with *Journalist* access to the *Journalist Interface* can:
  - Change the codenames associated with *Sources* within the interface.
  - Download, but not decrypt, submissions.
  - Delete one or more submissions.
  - Communicate with *Sources*.
  - If the *Journalist* has admin privileges on SecureDrop, they can create new *Journalist* accounts.

What a physical seizure of the *Journalist*'s property can achieve

- Tamper with the hardware.
- Prevent the *Journalist* from working on SecureDrop for some period of time.
- Access any stored, decrypted documents taken off the Secure Viewing Station.
- If the property is seized while powered on, the attacker can also analyze any plaintext information that resides in RAM.
- A physical seizure of, and access to, the *Journalist*'s Tails persistent volume, password database, and *Two-Factor Authentication* device will allow the attacker to access the *Journalist Interface*.

What a compromise of the *Application Server* can achieve

- If the *Application Server* is compromised, the system user the attacker has control over defines what kind of information the attacker will be able to view and what kind of actions the attacker can perform.
- An attacker with access to the **debian-tor** user can:
  - View, modify, and delete all files owned by this user. This includes sanitized Tor logs, created using the SafeLogging option, for SSH, the *Source Interface* and the *Journalist Interface*.
  - View, modify, and delete the Tor configuration file, root is required to reload the config.
- An attacker with access to the **ossec** user can:
  - Add, view, modify, and delete the log files, and in doing so send inaccurate information to the *Monitor Server* and the admin.
- An attacker with access to the **www-data** user can:
  - View, modify, and delete all files owned by this user. This includes all files in use by the SecureDrop application, such as text, code, the database containing encrypted submissions and communications. The attacker needs root access to reload configuration files.
  - View, modify, and delete both access and error logs for the *Journalist Interface*.
  - View any HTTP requests made by the *Source*, the admin, and the *Journalist* in that moment. This includes seeing plaintext codenames, submissions, and communications.
  - Add and delete communications between a *Journalist* and a *Source* by writing to the database.
- An attacker with access to the **root** user can:

  - Do anything the **www-data** user can do in terms of the SecureDrop application, this user is in full control of the server and can view, modify, and delete anything at will. This user is not able to decrypt submissions or communications, unless the attacker has access to the encryption key required to do so.

What a physical seizure of the *Application Server* can achieve

- If the *Application Server* is seized, the attacker will be able to view any and all unencrypted files on the server. An attacker will be able to modify any and all files on the server. This includes all files in use by the SecureDrop Application. If the server is seized while it is powered on, the attacker can also analyze any plaintext information that resides in RAM. The attacker can also tamper with the hardware.

What a compromise of the *Monitor Server* can achieve

- If the *Monitor Server* is compromised, the system user the attacker has control over defines what kind of information the attacker will be able to view and what kind of actions the attacker can perform.
- An attacker with access to the **debian-tor** user can:
  - View, modify, and delete all files owned by this user. This includes sanitized Tor logs, created using the SafeLogging option, for SSH.
  - View, modify, and delete the Tor configuration file, root is required to reload the config.
- An attacker with access to the **ossec** user can:
  - View all ossec logs and alerts on disk.
  - Modify the ossec configuration.
  - Send (or suppress) emails to administrators and *Journalists*.
- An attacker with access to the **root** user can:
  - Do anything the **ossec** user can do in terms of the SecureDrop application, this user is in full control of the server and can view, modify, and delete anything at will. This user is not able to decrypt encrypted email alerts, unless the attacker has access to the encryption key required to do so.

What a physical seizure of the *Monitor Server* can achieve

- If the *Monitor Server* is seized, the attacker will be able to view any and all unencrypted files on the server. This includes all files in use by OSSEC. If the server is seized while it is powered on, the attacker can also analyze any plaintext information that resides in RAM. The attacker can also tamper with the hardware.
- If the *Monitor Server* is no longer online or tampered with, this will have an effect on the quantity and accuracy of notifications sent to admins or *Journalists*.

What a compromise of the *Secure Viewing Station* can achieve

- The *Secure Viewing Station* is only useful to an attacker while powered on and with the Tails persistent volume mounted. The attacker may learn more if the *Transfer Device* or the *Export Device* are in use at the time of compromise or seizure. A physical seizure of this machine, its Tails device, the *Transfer Device* or the *Export Device* will also achieve nothing, assuming that the Tails and VeraCrypt implementations of full-disk encryption work as expected.
- A compromise of the *Secure Viewing Station* allows the attacker to:
  - Run commands as the **amnesia** user.
  - View, modify, and delete files owned by the **amnesia** user. This includes the *Submission Private Key* used to encrypt and decrypt submitted documents.
  - View, modify, and delete submissions in encrypted form

  - View, modify, and delete decrypted submissions, if they are stored in decrypted form on the *Secure Viewing Station*, or if the *Export Device* is in use.
  - Export the *Submission Private Key* key (unless there is a passphrase set).

What a physical seizure of the *Secure Viewing Station* can achieve

- The *Secure Viewing Station* is only useful to an attacker while powered on and with the Tails persistent volume mounted. The attacker may learn more if the *Transfer Device* or the *Export Device* are in use at the time of compromise or seizure. A physical seizure of this machine, its Tails device, the *Transfer Device* or the *Export Device* will also achieve nothing, assuming that the Tails and VeraCrypt implementations of full-disk encryption work as expected.
- A physical seizure of the *Secure Viewing Station*, while on and with the persistent volume decrypted and mounted, allows the attacker to:
  - Tamper with the hardware.
  - Run commands as the **amnesia** user.
  - View, modify, and delete the *Submission Private Key* used to encrypt and decrypt submitted documents.
  - View, modify, and delete decrypted submissions, if they are stored in decrypted form on the *Secure Viewing Station*, or if the *Export Device* is in use.

What a local network attacker can achieve against the *Source*, admin, or *Journalist*

- A local network can observe when they are using Tor.
- A local network can block Tor and prevent them from accessing SecureDrop.
- A local network may be able to deduce use of SecureDrop by looking at request sizes, plaintext uploads and encrypted downloads, although research suggests this is very difficult.

What a global adversary can achieve against the *Source*, admin, or *Journalist*

- A global adversary capable of observing all Internet traffic may have more luck than the local network attacker in deducing use of SecureDrop by looking at request sizes, plaintext uploads and encrypted downloads.
- A global adversary may be able to link a *Source* to a specific SecureDrop server.
- A global adversary may be able to link a *Source* to a specific *Journalist*.
- A global adversary may be able to correlate data points during a leak investigation, including looking at who has read up on SecureDrop and who has used Tor.
- A global adversary may be able to forge an SSL certificate and use it to spoof an organization's HTTPS *Landing Page*, thereby tricking the *Source* into visiting a fake SecureDrop site.

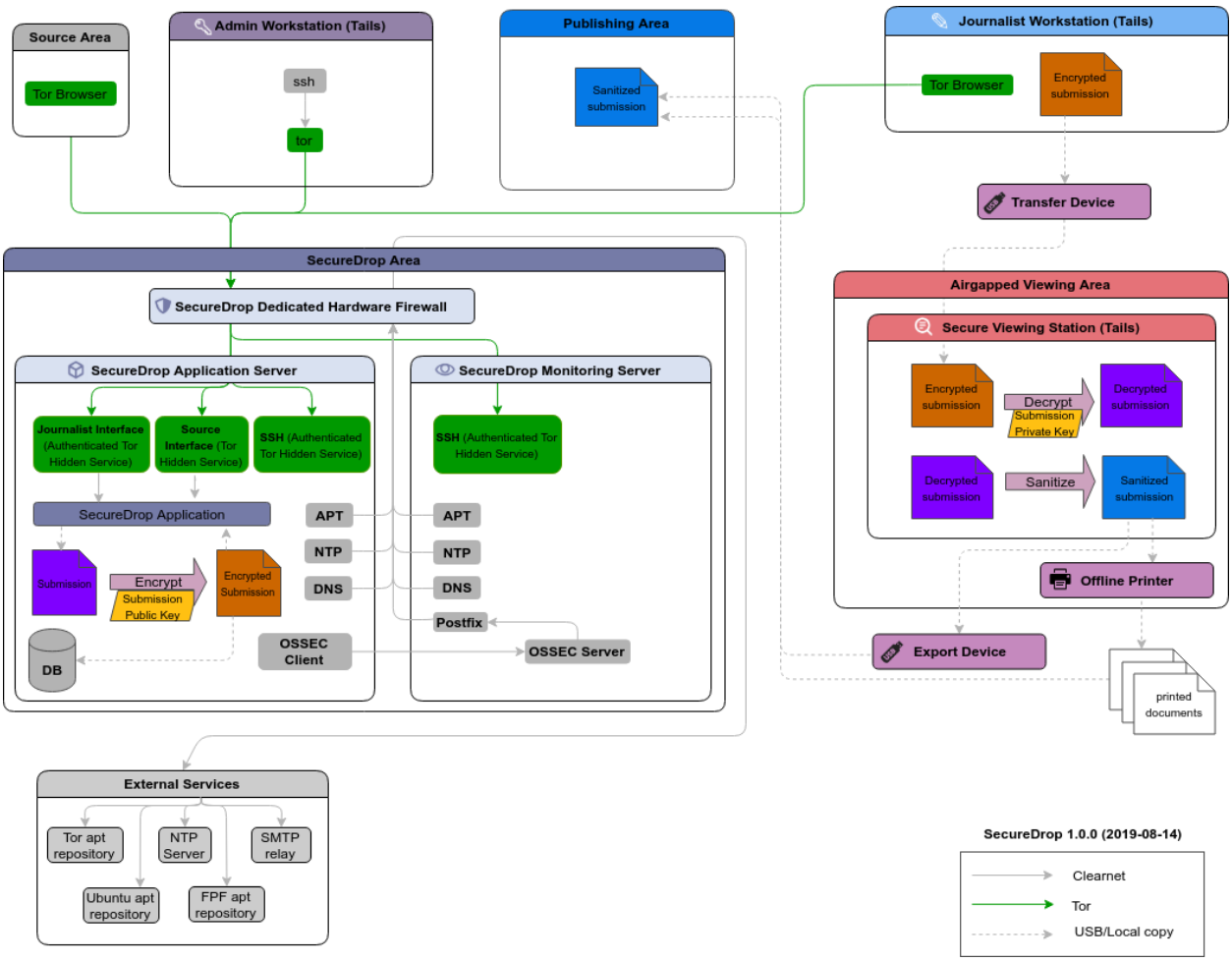
What a random person on the internet can achieve

- A random person can attempt to DoS the SecureDrop server and overwhelm the *Journalists* by generating a high number of codenames and uploading many large documents.
- A random person can submit empty, forged, or inaccurate documents.
- A random person can submit malicious documents, e.g. malware that will attempt to compromise the *Secure Viewing Station*.
- A random person can attempt to get sensitive information from a SecureDrop user's browser session, such as the *Source*'s codename.

- A random person can attempt to compromise the SecureDrop server by attacking the exposed attack surface, including the kernel network stack, Tor, Apache, the SecureDrop web interfaces, Python, OpenSSH, and the TLS implementation.

1.72 Data flow diagram

The following diagram captures all data flows to and from a SecureDrop deployment.



1.73 Attacks and countermeasures on the SecureDrop environment

SecureDrop is a complex ecosystem comprised of various pieces of hardware, a diverse codebase, multiple user roles, and varied software dependencies. As such, an adversary can compromise any one of these components through a variety of attacks, as detailed below. We've categorized attacks and countermeasures by SecureDrop architecture area for clarity.

There are certain attacks that cannot be mitigated by any of the technical or operational countermeasures built into SecureDrop. Attacks of a political nature — for example, if a *Source*, *Journalist*, or organization is threatened with legal action — are context-dependent, and determined by an ever-shifting climate around press freedoms. While these attack vectors are out of the scope of this document, they should be factored in to any organization's threat model with regional and political specificity.

1.73.1 Application code — SecureDrop repository/release

Attacks to the application code — SecureDrop repository/release

- Malicious code introduced in SecureDrop repository
- Malicious code introduced in SecureDrop release
- Failure to encrypt submissions as they are written to disk

Countermeasures on the application code — SecureDrop repository/release

- Code (git tags) and releases (packages uploaded to apt) are signed with the airgapped signing key
- Protection is placed on *main* and *develop* branch on GitHub
- For SecureDrop Developers, *Two-Factor Authentication* is mandated on GitHub
- Community trust is built through 3 trusted code owners and code reviews

1.73.2 Application code — *Source Interface* and *Journalist Interface*

Attacks to the application code — *Source Interface* and *Journalist Interface*

- Configuration vulnerability in *Source* or *Journalist Interface*
- Lack of segmentation between *Source* and *Journalist Interface*
- Session management vulnerability in *Source* or *Journalist Interface*
- Malicious input vulnerability in *Source* or *Journalist Interface*
- Configuration vulnerability in *Source* or *Journalist Interface*
- Authentication vulnerability in *Source* or *Journalist Interface*
- Access control vulnerability in *Source* or *Journalist Interface*
- Data protection vulnerability in *Source* or *Journalist Interface*
- Communications vulnerability in *Source* or *Journalist Interface*
- Error handling and logging vulnerability in *Source* or *Journalist Interface*
- HTTP security configuration vulnerability in *Source* or *Journalist Interface*
- File and resource vulnerability in *Source* or *Journalist Interface*
- Business logic vulnerability in *Source* or *Journalist Interface*
- Web services vulnerability in *Source* or *Journalist Interface*

Countermeasures on both *Source* and *Journalist Interfaces*

- *Interfaces* run on an end-to-end encrypted Tor *Onion Service*
- Sensitive source and submission data is sent through HTTP POST
- All source submissions are encrypted with GPG at rest using the airgapped *Submission Key*
- *Interface* sessions are invalidated after a user logs out or inactivity over 120 minutes
- Session control on *Interface* includes CSRF token in Flask Framework
- All *Interface* session data (except language and locale selection) is discarded at logout, and fully deleted upon exiting Tor Browser

- A number of mitigations are in place as protection against malicious input vulnerabilities on the *Source* and *Journalist Interfaces*:
  - X-XSS-PROTECTION is enabled
  - Content-Security-Policy is set to "default-src 'none'; script-src 'self'; style-src 'self'; img-src 'self'; font-src 'self';"
  - SQLAlchemy is used as ORM for all database queries
  - Application does not execute uploaded submission data
- A number of mitigations are in place as protection against the risk of an HTTP misconfiguration on the *Source* and *Journalist Interfaces*:
  - Cache control header is set to "no store;"
  - HTTP headers do not expose version information of system components
  - X-Content-Type is set to "nosniff;"
  - Content-Security-Policy is set to "default-src 'none'; script-src 'self'; style-src 'self'; img-src 'self'; font-src 'self';"
  - X-XSS-Protection is set to "1"

Countermeasures unique to *Source Interface*

- TLS on *Source Interface* is opt-in with an EV cert
- Only HTTP GET, POST and HEAD methods are allowed
- A number of mitigations are in place as protection against access control vulnerabilities on the *Source Interface*:
  - Source codenames are long and automatically generated
  - Hashed codenames are stored in a database hashed with a unique salt
  - Source codename reset functionality is not available
  - Source login does not display information about prior submissions
  - Source login requires 7-word codename to check *Source Interface* for replies

Countermeasures unique to *Journalist Interface*

- *Journalist Interface* is located behind an authenticated *Onion Service* and only privileged users have required authorization token
- Only HTTP GET, POST, HEAD and DELETE methods are allowed
- A number of mitigations are in place as protection against access control vulnerabilities on the *Journalist Interface*:
  - Apache autoindex module is disabled
  - *Journalist*/administrator passphrases are long and automatically generated
  - Passphrases are stored in a database hashed with a unique salt
  - Account generation/revocation/reset is restricted to Admin role
  - *Two-Factor Authentication* is required (via a TOTP app, or an HOTP device like a YubiKey)

1.73.3 *Application Server* and *Monitor Server*

Attacks on the *Application Server* and *Monitor Server*

- *Application* or *Monitor Server* configuration error
- *Source* or *Journalist Interface* is framed
- *Application* or *Monitor Server* is compromised
- Attacker exploits postfix
- Known vulnerabilities in the Linux kernel or packages used by the *Application* and *Monitor Servers*

Countermeasures on both *Application* and *Monitor Servers*

- Grsecurity/PaX linux patches prevent the exploitation of certain memory-corruption attacks
- AppArmor profiles further reduce process capabilities through Mandatory Access Control
- All SecureDrop infrastructure is provisioned via infrastructure-as-code (Ansible scripts)
- A cron job ensures that automatic nightly security updates are applied for OS packages
- *Journalist Interface* uses ATHS cookie
- *Monitor Server* should only expose SSH via Tor *Onion Service*. All other traffic should be blocked by firewall

Countermeasures unique to *Application Server*

- SecureDrop *Source* and *Journalist Interfaces* uses X-Frame-Options: DENY header
- Browser Same Origin Policy should prevent the SecureDrop page from trivial modifications, but more complex attacks are mitigated via the X-Frame-Options: DENY HTTP header

Countermeasures unique to *Monitor Server*

- OSSEC is used for intrusion detection/file integrity monitoring, and are sent to Admins via end-to-end encrypted email

1.73.4 SecureDrop dependencies — Python, Tor, Linux Kernel, apt, Qubes, Ubuntu, or hardware firewall vulnerabilities

Attacks on SecureDrop dependencies

- Known vulnerabilities in Python or libraries used by SecureDrop
- Known vulnerabilities in Tor (incl. *Onion Service* cryptography, authentication)
- Malicious apt package installed at install-time or during updates
- Known weakness in *Onion Service* cryptography
- GitHub is compromised
- Firewall is not up-to-date
- Qubes ISO malicious
- Ubuntu ISO malicious
- Tor apt repo compromised
- Ubuntu apt repo compromised
- Tor Browser exploit

- Vulnerabilities/Compromise of Hardware Firewall

Countermeasures against vulnerabilities in Python or libraries

- FPF performs vulnerability management for all Python packages used by SecureDrop
- CI will run safety check to ensure dependencies do not have a CVE associated with the version

Countermeasures against vulnerabilities in Tor

- A cron job ensures that automatic nightly security updates are applied for OS packages, including Tor
- Grsecurity/PaX linux patches prevent the exploitation of certain memory-corruption attacks
- AppArmor profiles further reduce process capabilities through Mandatory Access Control
- *Onion Service* authentication is used as a complementary authentication and only used for defense-in-depth/attack surface reduction

Countermeasures against malicious apt installs

- apt does GPG signature verification of all packages as long as it's not explicitly disabled

Countermeasures against malicious Qubes or Ubuntu ISOs

- SecureDrop *Admin Guide* instructs Users/Admins to validate checksum/signatures of downloaded images

Countermeasures against vulnerabilities in the hardware firewall

- SecureDrop *Admin Guide* informs administrators to update the hardware firewall and provides a very restrictive policy for accessing the administrative interface (blocked on app and mon ports of the firewall).
- Alert emails are sent out to admins when there are critical pfSense vulnerabilities.
- *Application* and *Monitor Servers* use IPTables as host-based firewall for defense-in-depth
- All application traffic is over Tor *Onion Services* (end-to-end encrypted) and all software packages are signed. Only DNS and NTP are transmitted over HTTP (unauthenticated and in cleartext)

1.73.5 Network Infrastructure — FPF Infrastructure or Organization Corporate Network

Attacks on network infrastructure

- *Landing Page* source control is compromised
- *Landing Page* host is compromised
- *Landing Page* is framed or unavailable
- *Landing Page* DNS leaks from SecureDrop/leaks-related subdomain
- Communications vulnerability in *Source* or *Journalist Interface*
- DNS requests to news organization's subdomain for SecureDrop *Landing Page*, Freedom.press, torproject.org Tor activity, SD submissions may be correlated
- SecureDrop.org is compromised
- User web traffic to SecureDrop *Landing Page* uses CDN and may be logged
- Tor network exploit
- apt server man-in-the-middle used to serve old or malicious packages

- SecureDrop apt servers are compromised, or apt server man-in-the middle attack injects malicious packages
- News Organization network is compromised
- OSSEC and/or Daily Journalist Alert SMTP account credentials compromised
- OSSEC and/or Daily Journalist Alert private key compromised
- SMTP relay compromised
- Admin's network is monitored

Countermeasures in FPF infrastructure

- Builds are independently validated by multiple developers
- Release files containing hashes (MD5, SHA1, SHA256, SHA512) of package file and package hashes are signed with an airgapped GPG key
- Developer key list is published and GPG-signed with the directory key
- SecureDrop updates are packaged in a .deb file and served through FPF's apt repo
- Source code is validated/verified before packaging and signing the .deb

Countermeasures in news organization corporate network

- SecureDrop environment should be strictly segregated from corporate environment
- Most SecureDrop traffic goes over Tor and as such is encrypted end-to-end
- Alert emails to *Journalists* and administrators are GPG-encrypted (but not signed) to provide confidentiality
- OSSEC alerts are scrubbed for sensitive contents (application data, server IPs)
- Documented deployment best practices provide instructions to strengthen *Landing Page* security and privacy

1.73.6 User Behavior and Hardware — SecureDrop Hardware Tampering or Failure in Operational Security

Attacks on user behavior or hardware

- *Journalist* corporate workstation seized/tampered/compromised
- *Export Device** seized/stolen/lost
- Admin *Two-Factor Authentication* device is lost or compromised
- Admin SSH Key is compromised
- SecureDrop installer misconfigures server/firewall hardware
- *Source* uses tor2web or employer/corporate device
- *Source* shares that they are using SecureDrop/leaking documents
- *Journalist*/administrator gets phished from a submission or otherwise breaks the *Secure Viewing Station* airgap with malware

Countermeasures in user behavior recommendations

- *Source Guide* gives instructions on best practices for the entire submission workflow
- *Source Interface* banner suggests that user disables JS (high security settings in Tor Browser)

- *Journalist Guide* informs users of malware risks, the importance of strict compartmentalization of SecureDrop-related activities
- *SecureDrop Deployment Guide* gives best practices for proper administration of the SecureDrop system, and its public-facing properties like the *Landing Page*
- *Admin Guide* gives instructions for long-term maintenance of the technical properties of the SecureDrop system, as well as operations to support *Journalists*
- All adminsitrator tasks are completed over Tor/Tor authenticated *Onion Services* after installation
- Any journalist/admin password/2FA credentials resets can only be done by an administrator with password-protected SSH capability or authenticated *Onion Service* credentials.

CHAPTER

TWO

GET INVOLVED

SecureDrop is an open source project. If you would like to contribute to SecureDrop, please see our developer documentation.

Two versions of this documentation are available, and can be selected in the lower left corner using the version dropdown menu:

- `latest` - built from the `develop` branch of the SecureDrop repository, containing updates that have been tested but not yet released.
- `stable` - built from the `stable` branch of the SecureDrop repository, and up to date with the most recent release, 2.14.0.