

SECURITY MONITORING AND TECHNICAL INFORMATION



MONITORING & HARDENING OF LUCY SERVERS

The following information describes the process of installation and support of a new *LUCY* server provided by Hetzner, Hosttech or Exoscale. In USA we work with Digital Ocean.

Installation and configuration of LUCY software

Installation of LUCY software is initiated by the Support team and it is performed in automatic mode. Support team purchases a new domain name for LUCY and manually configures internal E-mail server. After the installation Support team makes sure of correct functioning LUCY software by starting a test campaign and analyzing the campaign results.

After the purchase, the provider creates a new server that is ready for installation of LUCY software. The server can be provided by LUCY or the client (on premise or in the cloud). More info here: https://wiki.lucysecurity.com/doku.php?id=lucy_onboarding_checklist#technical_planning

Installation and configuration of protection tools

Protection software is necessary to provide continuous and secure functioning of LUCY servers.

All security software is distributed, installed and configured using *Playbook Ansible* in automated mode. Centralized installation and configuration of the security software allows us to achieve the unified automation for rapid installation and configuring of any server. After the installation of the protection tools, the system check is performed using *Lynis* tool, in case of successful system check, the server is completely ready for using by the Client.

LUCY server's protection tools

1. All LUCY servers have configured *Firewall* to restrict access to the servers. The access is only allowed for the System Administrator and Support team. The server can only be accessed through the Privileged Access Management server (XTAM):
Create Secure Remote Sessions;
Record Video Sessions for Playback;
Keystroke and Clipboard Text Event Recording;
Join in Real Time or Terminate Active Sessions;
Review Session Audit Logs for Compliance.
2. *Fail2ban* daemon is running for protection from brute-force attacks, it is configured to protect both SSH and Postfix.
3. *Auditd* daemon provides the detailed information about all system events, especially information on security violations that allows to take necessary actions. The event information is available in log files stored locally.
4. *Lynis* – a flexible tool that is normally executed after installation of a new server and allows to check a new system in the following ways:

SECURITY MONITORING AND TECHNICAL INFORMATION



- Security audits
- Compliance testing
- Penetration testing
- Vulnerability detecting
- System hardening

1. *Rkhunter* – is executed weekly, it is used to scan the server for rootkits, backdoors and possible local exploits. The scanning results are available in log files stored locally.
2. *Zabbix agent* – is used for monitoring processes and hardware on the LUCY server.
3. *Backup script* – is used for encrypting LUCY backups and transferring backups to the backup server.

Monitoring

Properly configured *Zabbix* agents are running in 24x7 mode on all LUCY servers, they are used for monitoring processes and daemons required for successful operation of LUCY. *Zabbix* agents keep connection to central stand-alone *Zabbix* server and report all important events to it. *Zabbix* server send alarm e-mails to System Administrator and to Support team in case of problems. Based on monitoring alarms System Administrator or Support team immediately react to any problem found, e. g. server crash, network unavailability, insufficient disk space, etc. In case of installation LUCY on a dedicated server with RAID *Zabbix* agent also monitors the status of the RAID array in order to prevent possible data loss due to hard disk malfunctioning. Elk stack and Wazuh software allow real-time monitoring of possible vulnerabilities in the used LUCY components. Elk stack and Wazuh software are also used for certain incident types. Based on the incident type, specific rules for escalation process are defined.

Elk stack and Wazuh software allow real-time monitoring of possible vulnerabilities in the used LUCY components. Elk stack and Wazuh software are also used for certain incident types. Based on the incident type, specific rules for escalation process are defined.

Automated Incident management

Elk stack and Wazuh software are also used for certain incident types. Based on the incident type, specific rules for escalation process are defined.

Backup

We backup all our LUCY servers on an external stand-alone backup server in encrypted form on a daily basis using our *Backup script*. If you do not require external backups, we can disable this feature.

Live Vulnerability Patching kernel

We use KernelCare to continuously deliver kernel security patches.

OS Patches

Patches to the OS (Debian) are automatically rolled out using the software “Patchman”

SECURITY MONITORING AND TECHNICAL INFORMATION



LUCY App updates

Since updating to the new version of LUCY is only possible when all campaigns are stopped, communication with the Client is necessary and the updating process is done manually after publishing a new LUCY version. The update can be done with a click of a button: https://wiki.lucysecurity.com/doku.php?id=update_lucy

Security & hardening inside LUCY app

Beside the backend patching/monitoring and hardening the LUCY admin can also harden the system within the admin interface (browser). The following options are available:

- Restricted network based access to LUCY: https://wiki.lucysecurity.com/doku.php?id=firewall_security_settings
- Secure the user access: https://wiki.lucysecurity.com/doku.php?id=password_policies_login_protection_strong_authentication
- 2FA Authentication: https://wiki.lucysecurity.com/doku.php?id=2-factor_authentication_2fa
- SSO: https://wiki.lucysecurity.com/doku.php?id=sso_authentication
- Restrict the user access: https://wiki.lucysecurity.com/doku.php?id=user_management
- Secure Setup of LUCY in DMZ: https://wiki.lucysecurity.com/doku.php?id=setting_up_a_lucy_master_slave
- Secure (and anonymous) storage of data: https://wiki.lucysecurity.com/doku.php?id=confidentiality_of_campaign_data
- Delete sensitive data: https://wiki.lucysecurity.com/lib/exe/fetch.php?media=lucy_monitoring_and_security.pdf
- Secure communication channels: https://wiki.lucysecurity.com/doku.php?id=ssl_configuration
- Transparent network communication: https://wiki.lucysecurity.com/doku.php?id=network_communication_-_lucy_-_internet
- Secure support: https://wiki.lucysecurity.com/doku.php?id=remote_ssh_support
- Updates of the application: https://wiki.lucysecurity.com/doku.php?id=update_lucy
- Custom Admin URL creation: https://wiki.lucysecurity.com/doku.php?id=custom_admin_url_title
- Ability to monitor all system activities: https://wiki.lucysecurity.com/doku.php?id=log_files_in_lucy
- Ability to monitor the system in real time: https://wiki.lucysecurity.com/doku.php?id=system_performance_monitoring
- Data Security, Processes Privacy (GDPR): https://wiki.lucysecurity.com/doku.php?id=company_application_and_data_security
- Ability to store key on HSM: https://wiki.lucysecurity.com/doku.php?id=company_application_and_data_security



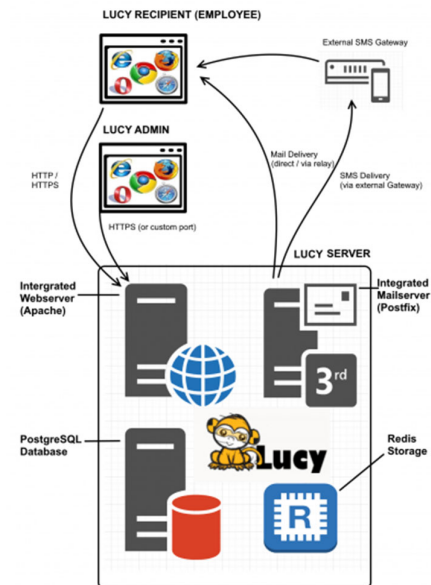
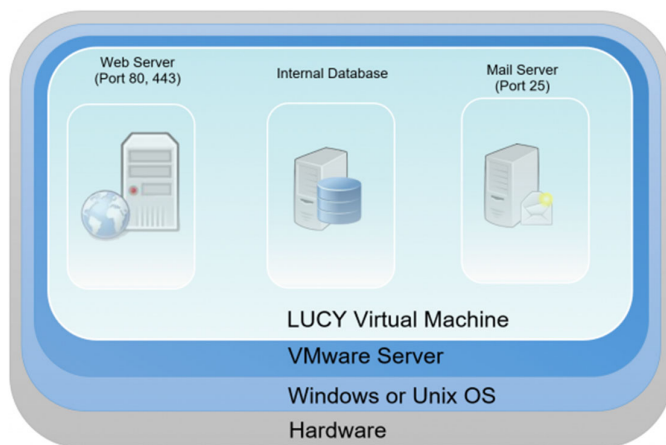
SECURITY MONITORING AND TECHNICAL INFORMATION

Technical info

This page describes the technologies and software used by LUCY. The information herein applies mostly to VM-based LUCY distributions (VMware Workstation, VMware ESXi, VirtualBox and Amazon) as LUCY installations on custom dedicated servers may have different configurations. The LUCY server contains different components. Those components are already installed when using our images (Vmware/AMI etc.) or are installed when using the Linux installer.

LUCY Vmware technical components

When you download and boot the VMware Image, all software components are integrated in that image. There is no need to install any additional software. All components (DB, mail server, web server etc.) are bundles within the VMware images and controlled by the internal LUCY software, which runs transparently in the background. The updating of those components is also done within the LUCY software through internal processes, which are not visible to the end user.



Browser & OS support

The list of supported browser and OS can be found [here](#).

Hardware requirements

The hardware requirements may vary depending on the amount of recipients you upload for testing, your campaign type and also how you run the campaign. More info [here](#).

SECURITY MONITORING AND TECHNICAL INFORMATION



External service API's

LUCY offers different API's to third party services or via REST API:

- URL Shortening services
- DynDNS services
- External mail relays (Sendgrid)
- SMS (Messagebird)
- Domain (Godaddy)
- LUCY REST [API](#)

Network communication

LUCY may require certain communication channels to servers on the internet. More info [here](#).

Operating System

Starting from 4.3, LUCY is running on a 64-bit **Debian 9 (Stretch)** system. There are no system patches or hardenings applied - LUCY uses a vanilla Debian distribution without any additions. The system is configured to download updates and new packages from a custom LUCY apt mirror, which has the same IP address, as LUCY license server (make sure it is open on your corporate firewall). The operating system gets updated only when you upgrade LUCY to a new version.

Accounts

There are a few accounts on the system like "phishing" or "support". The "phishing" account is the one that is used for Lucy file permissions. Also you can SSH to Lucy using "phishing" account and it will automatically launch the console setup program. The "phishing" account is required for Lucy to function properly, so it's not recommended for removal. If you SSH to the "phishing" account the console setup program ([python setup script](#)) is launched (with elevated privileges). So it can't be used as a full-featured SSH login. The purpose is to only run the setup console.

The "support" account is used to log into the system over SSH when user turns on SSH access for support. It's safe to remove the "support" account, but then we won't be able to log into that system over SSH for support purposes.

Web Server

Lucy web interface uses **Apache 2.4** as a web server. The server utilizes "mod-security" and "mod_headers" modules to hide underlying software signatures from external visitors.

SECURITY MONITORING AND TECHNICAL INFORMATION



Database

LUCY stores all related data in **PostgreSQL 9.6** RDBMS. All sensitive information stored in there is encrypted as PostgreSQL is available only for internal connections. There are no configurable options for the DB encryption. The encryption is mandatory for all data and is performed automatically with the following settings:

- It's a column-level encryption performed on both the application and DB layers before storing any data in the database. We don't use TDE (transparent database encryption), as PostgreSQL doesn't support it, so we encrypt only a subset of columns in DB – everything that holds client/attack/recipient-related data.
- We mostly perform the encryption/decryption on the application level, but there are certain queries that decrypt data on the DBMS level for convenience – for sorting & data search.
- The encryption is performed using AES-256-CBC.
- On demand we can provide a HSM solution, that will allow us to use a HSM-based encryption – in that case the encryption key will be stored on the external hardware module with anti-tampering protection.

Intermediary storage

LUCY also uses **Redis Server 3.2** as an intermediary storage (task queue) when passing input data and results between users and background system workers. The data stored in Redis is not encrypted.

Software, secure coding & hardening

LUCY code is a **PHP** application based on **Yii Framework**. Certain parts of the system (some background scripts) are implemented using the Python programming language. We follow the following security principles:

- OWASP Top 10 / 2017
- Framework-level SQL injection prevention
- Framework-level CSRF prevention
- Lucy partially conforms to "CIS Debian 8" checklist (50% conformance: we can provide a detailed list of non-conforming items upon request. There are no critical issues in uncovered parts)
- Lucy partially conforms to "CIS PostgreSQL 9.5" checklist (50% conformance: we can provide a detailed list of non-conforming items upon request. There are no critical issues in uncovered parts)

SECURITY MONITORING AND TECHNICAL INFORMATION



Folders

Almost all LUCY files are located in `/opt/phishing` folder, which contains the system code, user files and settings. Normally, you shouldn't have to deal with LUCY code, so the most useful directories are those where user files are stored. All user-related files are placed in `/opt/phishing/files` folder:

- Attachment-templates — File templates for file-based attacks (exes and macro files)
- Awareness-templates — Awareness template storage
- Campaigns — Campaign and scenario files storage
- Domains — Domain DKIM configuration storage
- Header-images — Header images for report templates
- Page-templates — System page templates (for example, 404 page template)
- Recipient-groups — Recipient group storage
- Report-templates — Report templates file storage
- Scenario-templates — Scenario templates file storage
- System — Keeps system files: Custom logo, system SSL certificates and system-wide static files for landing pages

Logs

There are several places in the system where you can find LUCY-related logs. They can be helpful to resolve or diagnose issues using your own technological resources without involving LUCY support. First of all, you should look at Apache web server logs directory `/var/logs/apache2`, where web server saves access and Error log files:

- access.log
- error.log

The next place with logs is `/opt/phishing/runtime`, where LUCY application logs are stored:

- application.log — Web application log - here you can find all web interface error notifications.
- beef.log — BeEF vulnerability detection framework log.
- console.log — Console commands log - Contains errors and issues for periodical background commands.
- resque_system.log — System background tasks log.
- resque_worker.log — Service background tasks log - Almost all background tasks you use in LUCY (including sending emails, copying websites, etc.) write logs to this file.
- scheduler.log — Scenario scheduler log.

SECURITY MONITORING AND TECHNICAL INFORMATION



Critical Services

There are several system services that are critical for LUCY. You can check if they are running by executing "ps aux" command. If some of required services are not running, then they should be started using "service NAME start" command (where NAME is the name of the service you are going to start):

- apache2
- postgres
- redis-server
- supervisor

Frequently asked questions (FAQ)

Which components are installed with the installer script? What modifications are made to Linux OS by the installer?

When Lucy is installed as a docker image, it gets downloaded from <https://hub.docker.com> (you can search for Lucysecurity there) to internal docker storage that is managed by docker itself – as any other usual docker image. Usually it's `/var/lib/docker/`. After that the installer launches the image with specific parameters, extracts a management script (that launches Lucy's console setup tool) to `/tmp/files`, moves it to `/usr/bin/lucy-setup.sh` and removes the `/tmp/files` folder. These are all changes that the docker-based installer does, all other software and files are within the container.

When installing Lucy in a "legacy" way (which is possible only for Debians, as the installer requires some Debian-specific packages and software versions), the installer adds around 500 new packages, does very deep system changes, including configuring the login shell, changing other services (postfix, apache, etc).

The installer checks if docker is installed and installs it from the official docker website. Are there any downsides of having docker engine pre-installed?

No, pre-installed docker engine is just fine and will work completely the same. Just make sure it will be updated on time.

Who patches the Docker Runtime and host Linux system when security vulnerabilities become known?

When using a Docker-based installation, the Docker engine and all packages on the host are managed by the host package management system – be it RHEL package management or Debian/Ubuntu APT. Lucy's docker container has no access to outer system and therefore cannot install any updates there. When using a "legacy" installation, all software packages are updated along with Lucy updates using Lucy's Debian mirrors.

SECURITY MONITORING AND TECHNICAL INFORMATION



How does Lucy update when it's running in Docker?

After installing a Docker container with Lucy, Docker's container system is not involved anymore - all updates are performed within the container using container's APT system with our custom package mirror.

Is the container included in the installer payload or is the container loaded via docker pull?

Lucy's container is downloaded from <https://hub.docker.com> (official Docker Hub) during installation.

When configuring proxy in Lucy, what software is affected?

In case of Docker installation, only software inside the container is affected. There are no consequences for the software on the host system. When using a legacy installer, the proxy configured will be used as a default proxy for the whole Linux system.

How do you back up Lucy?

Speaking of Docker container installation, the backup is seamless and is a piece of cake. First of all you "commit" the container, making a static image of it with all files, changes, etc: **docker commit lucy lucy-backup** After that you:

- either save it as a tar file: **docker save -o /path/to/lucy-backup.tar lucy-backup**
- or export it to your private docker registry: **docker push lucy-backup**

The backup image can be recovered easily from file by **docker load -i /path/to/lucy-backup.tar** (in case of local file) or **docker pull lucy-backup** (in case of private docker registry)

After that you can start your container as a regular docker container with this command: **docker run --privileged -v /proc/sysrq-trigger:/sysrq -d -p 80:80 -p 443:443 -p 25:25 -p 5001:5001 --name lucy --restart=always lucy-backup /bin/bash /etc/run-services.sh**

Things get a bit more difficult in case of "legacy" installation - you should back up multiple directories, where configurations and files are stored:

- /opt/phishing
- /etc/
- /var/lib