VPS HARDENING

Please find a complete documentation here:

lucy_monitoring_and_security.pdf

All LUCY serves 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 **PAM** (Privileged Access Management server).

Fail2ban daemon is running for protection from brute-force attacks, it is configured to protect both SSH and Postfix.

Auditd daemon provides detailed information about all system events, especially information on security violations that allows taking necessary actions. The event information is available in log files stored locally.

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

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.

Zabbix agent – is used for monitoring processes and hardware on the LUCY server.

Backup script – is used for encrypting LUCY backups and transferring backups to the backup server. We back up 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.

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.

Live Vulnerability Patching kernel. We use KernelCare to continuously deliver kernel security patches.

“Patchman” deliveries patches to the OS (Debian) automatically.

SSH root access can be provided by request, we will need your public ssh-rsa key. Access with a password is not allowed.

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 9" 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)
- PHP 5.6 is being updated using mirrored repo, which contains security patches developed by Microsoft. Source of the patches.

OS

Starting from 4.3, LUCY is running on a 64-bit Debian 9 (Stretch) system. 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 or if used with a VPS automatically.

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.

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.

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

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

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.
- 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.