System Description and Risk Analysis

Francesco Berla   Mose Mizrahi Erbes   Etienne Salimbeni
Jan Schär

2021-12-02

Contents

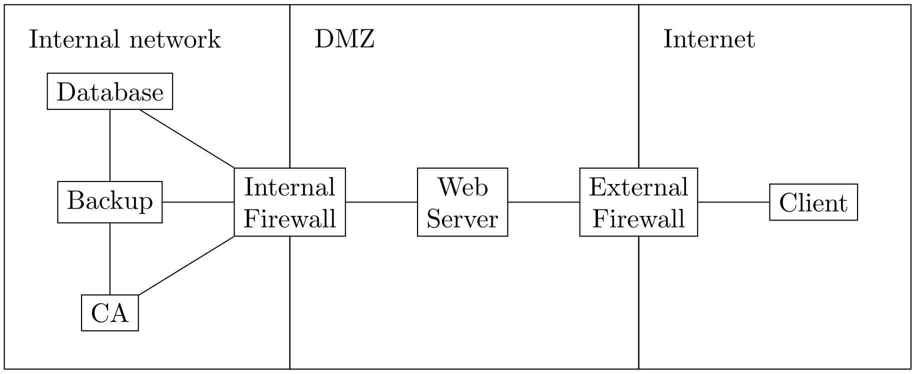

Figure 1: Network diagram

1 System Characterization

1.1 System Overview

The system is composed by seven components:

Client It will allow the end user to connect to the system.

External Firewall Filters the traffic from the client and the outside world to the web server.

Internal Firewall Filters the traffic from the web server (the DMZ) to the internal network of the organisation (ca, backup, database).

Web Server Serves the logic of the application, validating and orchestrating all the requests coming from the client.

Database Stores user information, including passwords.

CA Creates, signs and stores the certificates that will be used by the users. Revokes certificates, and returns a CRL.

Backup Stores the data stored in the database and the CA, stores logs, and stores all the configurations needed to recreate the whole system in a secure way in order to restore complete functionality in the event of a failure or compromise.

All servers are connected through two local networks. Only the external firewall is directly reachable from the internet. The internal components (ca, backup, database) are accessible only through the internal firewall. In Figure 1, the networks and connections during operation are shown. For administration, the client can connect to any server via the firewalls (using ssh jump).

1.2 System Functionality

The system is intended to secure the emails exchanged inside the company and with informants. The objectives are to have emails that cannot be read from unauthorized parties, cannot be altered in any way and have the guarantee of the identity of the sender.

The system should keep track of the issued and revoked certificates in order to maintain the security guarantees mentioned above.

1.2.1 Certificate issuing

A user can authenticate with the web application using username and password, or a certificate. Once authenticated, he can navigate to the form for issuing a certificate, where he is prompted to choose a password. After submitting a form, the newly generated certificate, including the private key, is downloaded as a PKCS#12 file. The file is protected with the password which the user entered. The user has to enter the same password when importing the certificate into an application.

1.2.2 Certificate revocation

A user can authenticate with the web application using username and password, or a certificate. He can then access a table which lists all his certificates, and from here he can revoke individual certificates. Alternatively, the user can revoke all his certificates. The certificate revocation form asks for the revocation reason, which is then included in the CRL.

1.2.3 CA administrator interface

CA admins are a subset of regular users of the system. If a CA admin authenticates to the web application using a certificate, he can then access the CA admin interface. The CA admin interface displays the number of issued and revoked certificates, as well as the serial number of the last issued certificate.

1.2.4 Key backup

When a certificate is issued, it is stored in an sqlite database on the CA server, together with the private key. The private key is encrypted using the key backup public key. The key backup private key, which allows decrypting the backup keys, is stored offline. A python script is provided which performs this decryption. The sqlite database is regularly copied to the backup server.

1.2.5 System administration and maintenance

On the client machine, there is a sysadmin user account, which has an ssh private key. The corresponding public key is installed on all server machines. Thus, the system administrator can connect to any server using ssh for administration.

1.3 Security Design

1.3.1 Access control

Packet filtering The external firewall filters all communication between the outside world (including the client, which represents an external system) and the internal components, including the firewall itself. For TCP, only ports 22 (for SSH), 80 (for HTTP), and 443 (for HTTPS) are left open; packets sent to other ports are dropped. The client can use SSH to connect to any internal component. The client can only send HTTP/S packets to the web server, HTTP/S packets sent to other internal components are dropped. All UDP packets are dropped. ICMP is not blocked, the client can ping the web server. ICMP packets sent to other internal components are dropped. The internal firewall filters all communication coming from the external firewall and from the web server allowing only http/s, mysql, ssh and icmp to reach it. The internal firewall act as a NAT and as a single gateway to the internal components. The traffic is redirected based on the type to the right components (mysql traffic goes to the database, https goes to the ca and ssh is allowed on every host). Excluding ssh using public key authorization the backup is not accessible from outside the internal network.

Employee & CA administrator authentication Employees can access the employee interface at the URL "http://imovies.com", "https://imovies.com", "http://www.imovies.com", and 'https://www.imovies.com". They are always redirected to the https URL "https://imovies.com" if they visit any other URL. At this URL they are served a web page where they can authenticate themselves using a password or a certificate. If they authenticate with their username and passwords, then the web server checks if the database contains a username with a hash that matches the password's hash. If they authenticate with a certificate, the server checks if the certificate is signed by the CA, not expired, and not revoked. The revocation is checked with the CRL, which is always kept up-to-date. If they are authenticated, then they are served a web page where they can change their information, obtain new certificates, or revoke old certificates. If the employee is a CA administrator and has authenticated with a certificate, then they are also given a link through which they can access the CA administrator interface, where they can consult the state of the CA.

Database access The database employs access control using mysql role based access control. Only the user *web server* with traffic coming from the internal firewall can operate on the tables. The *web server* user is authenticated with a client certificate. With this mechanism we ensure that only the web server using the internal firewall as gateway is accessing the database and therefore we prevent any intruder machine in the network to access the tables even if they manage to be in the internal network or in the DMZ.

1.3.2 SSH security

Using SSH, the client can connect to any component. The SSH packets sent to the connected component will go through the firewalls. SSH will be used for system administration. The only authentication method for SSH will be public key authentication. This will avoid all security problems related to passwords (guessing, bruteforce, involuntary disclosure, leaks, ...). For the system reviewers to test SSH functionality, the client will have a user “sysadmin” with a private key with which it can authenticate itself to all the internal components. The private key will be protected with the passphrase “sshpass” and the keypair is based on elliptic curve asymmetric encryption with good parameters.

1.3.3 Session management

User sessions are handled with the library Flask-Login.

If the user authenticates with a password, then it is checked if the user id-password combination is valid. If so, then the user is given a session cookie, randomly generated with a secret key. The user can log out; if they do so they must re-authenticate themselves.

If the user authenticates with a certificate (TLS client authentication), then it is checked if the certificate is signed by the CA, not expired, and not revoked. If so, then the user authenticates according to the user id on the certificate. Since this verification happens on each TLS connection, there is no need for a session cookie. In order to log out, the user needs to remove the client certificate from the browser. A small utility is installed on the client which allows removing the certificate by clicking a log out button in the web interface.

1.3.4 Key management

The web server has a certificate for the public website and one for TLS client authentication with the CA and database. The CA and database both have a server certificate. All of these certificates are signed by an internal CA, which is separate from the main CA issuing certificates to users. The private key of this internal CA is kept offline.

1.3.5 Security of data at rest

Private keys are stored encrypted in the database, where the private key is stored offline.

The database backup is encrypted with asymmetric encryption, where the private key is stored offline.

In addition to that the backup machine is using RAID 1+0 in order to sustain failure of a part of the disks.

1.3.6 Security of data in transit

Connections from the client to the server, from the web server to the database, from the web server to the ca are encrypted and authenticated with TLS. For connections from the web server to the ca and to the database, the web server uses a certificate for TLS client authentication and the ca and database use a certificate for TLS server authentication, and the identity of the certificates are verified. Data transfers from the database and the web server to the backup are encrypted and authenticated using ssh.

1.4 Components

All server machines (all except the Client one) have a hardened Ubuntu 20.04 Server OS in order to simplify maintenance, therefore every mention of *hardened installation* in this section refer to system described in this paragraph. Ubuntu Server has been chosen because it is a Linux distribution known to have sane defaults (that would mitigate possible overlooks from the system administrators) and a very minimal set of pre-installed packages and active services. Historically Ubuntu has also been proactive in kernel security, deciding to deploy kernel security patches even before the mainline Linux kernel.

In addition, every server machine has a local minimal *nftables* Firewall configuration in a zero-trust network fashion with the objective to add another layer of protection. Every machine should have as few services and applications as possible in order to reduce the attack surface.

In order to improve compartmentalization [check if it is the right principle] we will take advantage of the Ubuntu integration of AppArmor and use it as an enhanced access control mechanism that it is simpler to configure compared to SELinux.

1.4.1 Client

OS Ubuntu 20.04, default installation.

User accounts employee, sysadmin.

Applications Mozilla Firefox, ssh client.

Data Records User certificate, private ssh key (in the sysadmin account)

Connections Connects only to the web server through the firewall, it does not have open ports or any other incoming connection other than the one with the web server.

1.4.2 External Firewall

OS Ubuntu Server 20.04, hardened installation.

Applications nftables 1.0.0, openssh 8.8p1-1.

Data Records Private ssh host key, configuration files, logs.

Connections It accepts connections from the outside world and connects only to the Web Server and the Internal Firewall (for ssh only).

1.4.3 Internal Firewall

OS Ubuntu Server 20.04, hardened installation.

Applications nftables 1.0.0, openssh 8.8p1-1.

Data Records Private ssh host key, configuration files, logs.

Connections It accepts connections only from the Web Server and from the External Firewall (for ssh only) and connects to the internal network composed by the CA, the Backup and the Database.

1.4.4 Web Server

OS Ubuntu Server 20.04, hardened installation.

Applications nftables 1.0.0, openssh 8.8p1-1, Python 3.8.10, Python web application.

Data Records Private ssh host key, client certificate/key for connection to ca/database, server certificate/key for public website, configuration files, logs.

Connections It accepts connections from the outside world that have been filtered by the External Firewall both on ports 80 and 443 for its role as web server and on port 22 for ssh maintenance, and it connects to the database and to the CA service for retrieving information and to the backup machine for storing configuration files but only through the Internal Firewall.

1.4.5 Database

OS Ubuntu Server 20.04, hardened installation.

Applications MySQL Community Server 8.0.27, openssh 8.8p1-1, nftables 1.0.0

Data Records Private ssh host key, server certificate/key for connection to database, configuration files, users information, certificate information, logs.

Connections It accepts connections from the Web Server and from the External Firewall (the only protocol allowed from the External Firewall is ssh, jumping through the Internal Firewall) that have been filtered by the Internal Firewall on port 22 for ssh maintenance and on port 3306 for serving the database content. It connects to the backup machine for storing configuration files and the database content.

1.4.6 CA

OS Ubuntu Server 20.04, hardened installation.

Applications openssh 8.8p1-1, nftables 1.0.0, Python CA application.

Data Records Private ssh host key, server certificate/key for connection to CA, configuration files, CA private key, sqlite database containing issued certificates with keys and revocation status, logs.

Connections It accepts connections from the outside world that have been filtered by the Internal Firewall on port 22 for ssh maintenance and on port 443 for serving the certificate services .

1.4.7 Backup

OS Ubuntu Server 20.04, hardened installation.

Applications openssh 8.8p1-1, nftables 1.0.0, [incremental backup application], [raid 10]

Data Records Private ssh host key, configuration files, log files, users information, certificate information.

Connections It accepts ssh connections from the outside world that have been filtered by both firewalls on port 22 for maintenance and on port X for accepting new backups (database data from database and configuration files from the web server). It accepts two types of ssh connections on port 22. The first one is standard system maintenance through ssh by the system admins that comes from the outside world after being filtered by both firewalls. The second type of ssh connection is from the webserver (after being filtered by the Internal Firewall), from the firewalls and from the database for backup transfer.

2 Risk Analysis and Security Measures

2.1 Assets

2.1.1 Physical Assets

Firewalls/Web Server/Database/Backup/CA server These are physical machines. They should be protected from physical access by unauthorized people, and could be vulnerable to physical damage. Physical integrity is important for these machines.

Local network connecting servers There are two local networks, the internal network and the DMZ. Both consist of an Ethernet switch, connecting the internal firewall, ca server, database server, and backup for the internal network, and the internal firewall, web server, and external firewall for the DMZ. Nothing else is connected to the network. Physical integrity is important.

Internet connectivity of External Firewall The Firewall is connected to the Internet via a second Ethernet port. The availability of the internet connection is important.

Storage disks of Firewalls/Web Server/Database/Backup/CA server The storage disks will store sensitive information. Physical integrity and confidentiality is needed for the storage disks, only system administrators should be able to access them. Availability is also important, particularly for the backup disks. There should be some backup disks that only the owner(s) of the company have access to.

2.1.2 Logical Assets

Software Assets

Web server nginx is used as a reverse proxy, and for serving static files. The web application itself is a custom built software written in Python.

Database The database is MySQL.

CA nginx is used as a reverse proxy, for handling TLS authentication. The CA application itself is a custom built software written in Python.

Client The client uses Firefox to access the CA interface.

Administration sshd is running on each server, and the admin uses an ssh client.

Information Assets

User information For each user, the user information includes the user's username, first/last name, email address, password, and whether the user is a CA admin. This information should be kept confidential and integrity protected. Confidentiality for the password and for whether the user is a CA admin is particularly important. Users should be able to access their names and email addresses, and modify their names, email addresses, and passwords. System administrators should be able to read all user information except passwords, and modify all user information including passwords. Availability is also important for user information.

Certificates Employees and CA administrators can download certificates for authentication, including the private key, in the PKCS #12 format. They should keep their certificates confidential and integrity protected. They can also revoke their certificates, and they should do so if the certificates get stolen so that the certificates ensure authenticity.

Private key of CA This is very critical, because it allows issuing arbitrary certificates. It needs to be protected against unauthorized use. At the same time, it needs to be available to the CA. The confidentiality and integrity of the private key is very important; only system admins should be able to read and modify the private key of the CA. The private key of the CA should always be kept available, since if it was lost, the employees would no longer trust the CA.

Private key of backup This is only needed in special circumstances. The confidentiality and integrity of the private key is very important. The private key should be only available to the owner(s) of the company. This private key should also never be lost, availability is very important for it.

The clock of the CA and the web server The CA should have a reasonably accurate clock so that it issues certificates with correct validity periods, and the web server should have a reasonably accurate clock so that it can accurately check for certificate validity. The integrity of the clocks should be protected. Only system administrators should be able to manually modify clocks.

Configurations The configurations of the machines are important for them to work correctly and securely. The configurations should remain available even if something goes wrong and some machines lose their configuration. Integrity is important for configurations, only system admins should be able to modify configurations. Confidentiality is also important, albeit to a lesser degree.

Certificate Revocation List The web server will download the certificate revocation list from the CA, and use it to tell if a certificate is revoked. This list is important in determining the authenticity of certificates. The availability and integrity of the CRL is important, only system administrators should be able to modify the CRL. Also, the CRL must be kept up-to-date.

Logs The log files should be kept confidential, integrity protected, and available. Only system administrators should be able to modify logs. Also, only system administrators should be able to read logs, since reading logs can help an adversary attack a system.

2.1.3 Persons

System Administrators A system administrator can gain complete access to any system via SSH. Thus it is critically important that they are not malicious, and that their SSH private keys are kept confidential.

Regular employees Regular employees can access the employee interface with a username and password or with a certificate. Employees can use the interface to change their passwords, revoke and download certificates for themselves, and to change their email addresses and their names. Employees can also view their emails. Informants providing investigative reporting information to the company are given employee accounts, as well as regular employees working for iMovies.

CA Administrators CA administrators are a subset of the employees. From the employee interface, they can do anything a regular employee can do, and in addition if they authenticate themselves with certificates they are given a link through which they can access the CA administrator interface.

2.1.4 Intangible Assets

Reputation of trustworthiness The company obtains information from informants, who will only share their information if they trust the company to keep it confidential.

Timeliness Informants expect to be able to revoke certificates immediately in case of compromise, and to be able to send and receive email at any time. This leads to a requirement of high availability of the revocation interface and the serving of the CRL.

2.2 Threat Sources

Programmers Programmers of the system might maliciously or accidentally program the system in a flawed way.

Nature Natural disasters might cause harm to the physical assets.

Component Failures The physical assets, such as backup disks, might fail.

Employees Relevant employees are the system administrators and the users of the system.

Script Kiddies Because the system is exposed to the internet, Script Kiddies are a potential concern.

Skilled Hackers Skilled Hackers may try to obtain sensitive information, manipulate information, or harm the availability of the system.

Competitors Competing movie companies might want to conduct espionage and read emails to extract valuable information.

Organized Crime Organized criminals might want iMovies movies to be not released, and use both hacking and blackmail to accomplish this goal.

Governmental Agencies Governmental agencies might want to cause harm if they disapprove of the movies made by iMovies. They might demand access to iMovies' data, and use this data in a lawsuit against iMovies.

Malware Both targeted and untargeted malware could be a potential concern.

2.3 Risks Definitions

Likelihood	
Likelihood	Description
High	The threat source is motivated and capable of attacking the system, and countermeasures are missing or ineffective.
Medium	The threat source is motivated and has limited capabilities, and countermeasures may not be fully effective in preventing attacks.
Low	The threat source is not motivated or not capable of attacking the system, or implemented countermeasures are effective in preventing attacks.

Impact	
Impact	Description
High	The event results in serious reputational harm and loss of trust to the company, reduction of the ability of the company to operate, or harm to informants.
Medium	The event results in considerable disruption to the operations of the company.
Low	The event results in moderate disruption to the operations of the company.

Risk Level			
Likelihood	**Impact**		
	Low	Medium	High
High	Low	Medium	High
Medium	Low	Medium	Medium
Low	Low	Low	Low

2.4 Risk Evaluation

2.4.1 *Physical Assets: Firewalls/Web Server/Database/Backup/CA Server*

No.	Threat	Countermeasure(s)	L	I	Risk
1	Component failures: The backup disks might fail	All data stored in the backup system is stored on four disks configured with RAID 10, which means that a single failed disk won't cause data loss.	*Low*	*High*	*Low*
2	Nature: Natural events such as a fire or an earthquake could damage the backup storage	There are no countermeasures; the backups are kept on a single physical computer.	*High*	*High*	*High*
3	Component failures: The firewalls, the web server, the database, and the CA server could fail. This could cause some data loss and lead to temporary unavailability	The backup system makes it so that only data created after the last backup can be lost. There is no countermeasure against unavailability in case the systems fail.	*High*	*Med*	*Med*
4	A physical attacker could break into the server room and damage the infrastructure, leading to a failure of the service	The server room is protected so that only authorized staff can access it.	*Low*	*High*	*Low*

2.4.2 *Physical Assets: Local network connecting servers*

No.	Threat	Countermeasure(s)	L	I	Risk
5	Physical network issues: The local network could crash due to problems with the Ethernet switch or with cables, disconnecting the internal systems and causing temporary unavailability	We retry connecting if there is a network failure. Other than that there is no countermeasure against this risk.	*Med*	*Med*	*Med*

2.4.3 *Physical Assets: Internet connectivity of Firewall*

No.	Threat	Countermeasure(s)	L	I	Risk
6	Physical network issues: The firewall's internet connection to the outside world could be disrupted by a physical issue, causing temporary unavailability.	We retry connecting if there is a network failure. There is no countermeasure against this risk.	*Low*	*Med*	*Low*

2.4.4 *Logical Asset: CA server*

No.	Threat	Countermeasure(s)	L	I	Risk
7	An attacker could get access to the server and disrupt the certificate service	We harden the server; we close all unused ports and services on the server that could be an open door for potential attacks	*Low*	*High*	*Low*
8	An administrator could voluntarily add a backdoor on the server	The implementation of the CA server should not be done by a single engineer	*Low*	*High*	*Low*
9	An administrator could create a certificate not using the standard protocol and validate it with the CA server, this could leak unwanted certificates that are not regulated by the company	All certificate creation should be monitored	*Low*	*Med*	*Low*
10	An attacker could corrupt the database within the CA server, if the database is incorrectly formatted the CA cannot work otherwise the CA server will serve corrupted certificates	Only allow the CA server itself to modify the database	*Low*	*High*	*Low*
11	An attacker could format a malicious request on the web server to perform an SQL injection on the CA server, in that case the attacker would have full access on the CA database and can corrupt its integrity	The CA server uses parameterized SQL queries, which prevents injection	*Low*	*High*	*Low*
12	The coders of the CA server could forget to consider an edge case or an exception, this would let the CA server crash if such an exception is raised	Most exceptions occur in the database queries and when receiving requests from the web server, use try catch in such cases	*Med*	*Low*	*Low*

2.4.5 *Logical Asset: Firewalls*

No.	Threat	Countermeasure(s)	L	I	Risk
13	Misconfiguration of the firewalls could lead to service disruption, connection failures, data loss and leaks	Validate the firewall settings though extensive testing and hire experienced programmers	*Low*	*Med*	*Low*
14	An attacker could get access to one of the machines in the internal network via a vulnerability in the firewall, which could to a loss of integrity	Carefully monitor the system, such as power consumption that could not be hidden by rootkits	*Med*	*High*	*Med*

2.4.6 *Logical Asset: Backup server*

No.	Threat	Countermeasure(s)	L	I	Risk
15	Misconfiguration of the the backup procedure could lead the backups being incomplete.	Not only one person should implement the backup server. The logs of the backup server should be checked regularly. Additionally, regular simulations of restoring from backups could be made to be sure that they are complete and working properly.	*Low*	*Med*	*Low*
16	The time interval between backups could be too large, which could lead to the backup missing important information.	The backup interval should be chosen in a sensible way and a new backup should be triggered if critical information is updated.	*Low*	*Med*	*Low*
17	An attacker could get access to the backup and alter its contents.	We harden the system, close all unused ports, shut down all unused services, and use firewalls. In addition to that all the data on the backup is encrypted with asymmetric encryption that can decrypted only using a private key stored offline. This reduces significantly the window of the attacker.	*Low*	*Med*	*Low*
18	An attacker could send fake logs to the backup in order to pollute them.	The logs are sent in encrypted channels from only authorized senders authenticated via public keys.	*Low*	*Med*	*Low*
19	An attacker could sniff the logs in transit in order to gain valuable information.	The logs are sent in encrypted channels.	*Low*	*Med*	*Low*

2.4.7 *Logical Asset: Web Server*

No.	Threat	Countermeasure(s)	L	I	Risk
20	Misconfiguration of the web server could lead to service disruption, data loss, leaks and the request of unwanted certificates	Validate the system though extensive testing and hire experienced programmers	*Low*	*Med*	*Low*
21	An attacker could get access to server, which could lead to a loss of integrity	Use a DMZ to protect the rest of the system from the main server, carefully monitor the main server, such as power consumption that could not be hided by rootkits	*Med*	*High*	*Med*
22	Network malware could compromise the server, disrupting its integrity	The experienced engineers setting up the system make sure the newest threats are handled by not using deprecated tools	*Med*	*High*	*Med*
23	An attacker could authenticate as a user if the user uses a very simple password that is easily guessable	We force the user to use passwords of at least 10 characters.	*Low*	*Med*	*Low*
24	An attacker could inject code into the main server, granting itself access	We sanitize all inputs of the user	*Low*	*High*	*Low*

2.4.8 *Logical Asset: Client*

No.	Threat	Countermeasure(s)	L	I	Risk
25	An attacker could execute a CSRF attack, submitting a form to change a user's information.	We protect all forms on the client with CSRF tokens.	*Low*	*High*	*Low*

2.4.9 *Logical Asset: Database server*

No.	Threat	Countermeasure(s)	L	I	Risk
26	A misconfiguration of the access control by an administrator could allow any user to access arbitrary information on the database	The firewalls don't permit database access to arbitrary users and the database itself has rules to allow access only to explicitly granted users from specific hosts.	*Low*	*High*	*Low*

2.4.10 *Persons: Administrators*

No.	Threat	Countermeasure(s)	L	I	Risk
27	An administrator is no longer available to maintain the system	Do not rely on a single person and invest time in detailed documentation	*Med*	*Med*	*Med*
28	An administrator voluntarily damage the system, disrupting the service or leaking secure data	Keep an eye on all administrators and ensure their interests are inline with those of the company	*Low*	*High*	*Low*

2.4.11 *Persons: Company Staff (not Administrators)*

No.	Threat	Countermeasure(s)	L	I	Risk
29	A staff member of the company can become an attacker voluntarily or not, and damage the server machines	Restrict the access to server to only system administrators	*Low*	*Med*	*Low*
30	A staff member can share his TLS certificate, making it possible for anyone to impersonate him	There is no countermeasure for this threat	*Low*	*High*	*Low*

2.4.12 *Intangible Goods: Customer Confidence*

No.	Threat	Countermeasure(s)	L	I	Risk
31	An attacker can publicly leak all certificates of the company, in that scenario the company will loose the trust of its customers	Secure the PKI with all the countermeasures listed in this report	*Low*	*Med*	*Low*

2.4.13 *Information Assets: Keys*

No.	Threat	Countermeasure(s)	L	I	Risk
32	Administrators can loose the ssh passwords and keys which could block them from managing the servers	More than one administrator is in charge of the keys	*Low*	*Low*	*Low*
33	An administrator can loose the backup key which would make loading the backup impossible	Secure the key on a separate disk	*Low*	*Med*	*Low*
34	An attacker can get access to the CA root key and certificate, due to an Administrator leaking, or a exploiting a vulnerability on the CA server, this would allow the attacker to create fake CRL and certificates	Add a short expiration date on the root CA key and certificate and monitor the logs of all servers for suspicious requests	*Low*	*High*	*Low*
35	An attacker can get access to the private key to secure the https connection between the servers, in the scenario all message can be altered and read	Permissions of private key files are set such that only the owner can access them, and separate programs run as separate users	*Low*	*High*	*Low*

2.4.14 *Information Assets: CA Database/Certificates/CRL*

No.	Threat	Countermeasure(s)	L	I	Risk
36	An attacker or an administrator could leak the CA database content, leading to a loss of confidentiality	Place the database on a local network, harden the server and implement strict access controls for the database	*Low*	*Med*	*Low*
37	An attacker inside the local network can listen to all communications, in that case the attacker can know all certificates issued for that moment	Use tls to encrypt communication between servers in the local network	*Low*	*High*	*Low*
38	An attacker can modify the content of the CRL, this grants him control over the revocation procedure	The CRL is not stored, but recreated at each request, this way it is not possible to modify the content of the CRL if not by intercepting it during communication.	*Low*	*High*	*Low*
39	A attacker manage to use a valid CRL to validate a certificate after a user revoking that certificate. This allow the attacker to use the certificate during that crl time validity period	Reduce the CRL ttl to small time intervals to reduce the time window attack of the attacker	*Low*	*High*	*Low*

2.4.15 *Information Assets: Database/User Credentials*

No.	Threat	Countermeasure(s)	L	I	Risk
40	An attacker or an administrator can leak the database content, in such a scenario the company loses its integrity for its service	Place the database on a local network, harden the server and implement a strict access control protocol for the database	*Low*	*Low*	*Low*
41	A user could lose login credentials, and thus need to change their user id or their password	The administrator can manually set a new password in the database.	*High*	*Low*	*Low*
42	An attacker could manage to get a user's credentials, and then login in as the user, change the user's password, and revoke/issue certificates on the user's behalf.	Monitor suspicious queries in the logs and invalidate an account when detecting this event.	*Med*	*Low*	*Low*

2.4.16 *Information Assets: Backup*

No.	Threat	Countermeasure(s)	L	I	Risk
43	An attacker can corrupt the backup data which could lead to the loading of false certificates if the backup is used.	All backups are moved to an high privileged folder making accessible only with root privileges and encrypted therefore a modification would make them unusable but no false certificate can be inserted.	*Low*	*Low*	*Low*

2.4.17 *Information Assets: Logs*

No.	Threat	Countermeasure(s)	L	I	Risk
44	An attacker or a misconfiguration in an application can delete or modify existing logs from a machine, this allows him to remove the signs he left to get in the system.	The logs are sent immediately to the log server in an append only way making the local modifications ineffective.	*Low*	*Med*	*Low*
45	An attacker can delete or modify existing logs from the log server, this allows him to remove the signs he left to get in the system.	All the logs are periodically removed from the log server in order to be encrypted and stored in a high privilege folder, this reduces the attack window to a very short time.	*Low*	*Med*	*Low*

2.4.18 Detailed Description of Selected Countermeasures

Web Server The webserver application runs as a systemd service, with many of the systemd sandboxing options enabled, e.g. most of the file system is read-only for the web application process (*minimum exposure*). All SQL queries use parameterized queries, and HTML is constructed using a templating language, which automatically escapes all inserted strings; this should prevent any HTML injection and XSS vulnerabilities.

CA The CA just implement the very few services it is required in order to follow the *simplicity* and *minimise exposure* principles. This is why a simple custom CA is used instead of a legacy one. The custom CA server also respects the *fail-safe default* and *traceability* principles in its design by handling all exceptions and the use of logs. Lastly the use of the *trustworthy* python cryptography library ensures the CA *secret generation*, the server only using random generators and key creation functions explicitly from that library.

Database The database is based on the legacy database given but some action have been made to improve its security: as for other components in the network is a dedicated machine with a single purpose (Simplicity, Minimum Exposure) and the mysql role based access control mechanism has been used to allow only the desired user from the desired host (the web server) to read and manipulate the tables in the database.

Firewalls Firewalls are implemented using state of the art software (nftable) that is tightly integrated with the Linux kernel and is in use and has been vetted by the largest companies in the world (Simplicity and Open Design). The rules are written in a white-list fashion with a default blocking policy with only certain traffic allowed (Secure, Fail-Safe Defaults). The firewalls are designed machines that are responsible only to act as firewalls and no other services is expected from them, thus reducing attack surface and simplifying the management (Minimum Exposure).

Backups Backups are centralized in a single machine that has no operations other than the reception and encryption of database dumps and logs. This allows us to have a very simple design and have a separation of concerns (compartmentalization). In addition to that the log server is a standard rsyslog server that allows us to have sane defaults. The logs and the dumps are transmitted from the other machines using ssh with public key authentication allowing us to have encryption, integrity and authenticity for all the files received. In order to secure the logs that the rsyslog server collects and the database backups that are sent, we move them to a root owned directory (therefore accessible only with root privileges) and encrypt them with an `age` public key. This allows us to not have unencrypted logs and databases laying around that could be read or altered by an attacker. The encryption is done with `age`, a new generation encryption tool that is the product of the cryptographic community. It has secure defaults, an open design, highly secure way of generating the key pairs (secrets) and it is extremely simple to use. Ssh, rsyslog and age are open source technologies and are very simple to use, have been vetted by many people and have been used for quite some time, this allow us to honor *Simplicity, Open Design, Secure Defaults and Usability.* In addition to those measures, there is a dedicated user only for transmitting the logs to the central server, this user cannot login and can only send the files to the central server. This allow us to employ the *Least Privilege* principle.

2.4.19 Risk Acceptance

No. of threat	Proposed additional countermeasure including expected impact.
2	We accept the risk and we try to mitigate the effects with insurance. We could introduce off-site backups in the future to reduce the risk of backup failure.
3	We accept the risk but we try to perform regular controls in order to minimize the likelihood. Having redundant physical machines could decrease this risk.
5	We accept the risk but we try to be ready to substitute the failing components.
14	We accept the risk, additional measures could be a rigorous patch policy and external audits.
21	Additional measures could be regular external security audits to our infrastructure and web server in order to minimize the vulnerabilities and therefore reduce the likelihood.
22	Additional measures could be a strict update policy to decrease the weakness to known vulnerabilities and user education in order to reduce the effect of phishing campaigns.
27	We accept the risk, and we will try to always keep multiple system administrators so that the system isn't dependent on a single person. In addition to that the company could have organisational policies that an administrator can leave only after having transferred his knowledge to the new one.