
# THRON Security and Data Management

Technical Documentation
Release number: 1.3
Date (mm/dd/yyyy): 31/05/2018




Headquarters: Via Dei Contarini, 5/A, Piazzola sul Brenta (PD), Italy P.IVA 03586990289 www.thron.com

Summary

This article is designed to highlight the main security features of THRON. It gives a high-level architecture description of the platform to allow security departments to plan risk assessment documents and security compliance reports.
This article aims to provide:

- insight on design and development practices used to create the product
- a detailed view on protocols used between various architectural elements
- a specific focus on operational procedures that THRON uses to grant availability of content and services.

Click **here** to download pdf version.

# 1 Development Standards

THRON is a modern software application based on a cloud computing architecture to provide a flexible, scalable and up-to-date service to all its users; it leverages the most powerful languages and technologies available today: it takes advantage of hybrid (oop and functional programming) languages to implement server-side scalable services as well as html5 and ajax to create a responsive and engaging client-side user experience.

## 1.1 Design principles and procedures

A defined workflow has been adopted while developing THRON, the next paragraphs give an overview of design principles and procedures used.
Security principles:

- Design and implement systems considering the security point of view;
- Provide mechanisms to grant confidentiality, integrity and non-repudiation for transmitted information;
- Provide mechanisms to allow a controlled access to systems involved;
- Provide intrusion detection and protection mechanisms;
- Protect information from unauthorized access;

- Monitor systems from a security perspective.

Security objectives:

- Prevent attacks to systems;
- Manage security attacks and threats;
- Guarantee access control to systems and information;
- Prevent unauthorized access to information and systems;
- Guarantee law compliance.

## 1.2 Secure coding

Many best practices have been adopted during THRON development, in the current paragraph we will present a list of the most relevant ones.

Input and output validation:

- All data returned outside application's trust boundary should be encoded (eg. SOAP, JSON);
- Encode all characters unless they are known to be safe for the intended interpreter;
- Require authentication before a file is uploaded;
- Limit file types which can be allowed for uploads;
- Do not save files in the same context as web application;
- Ensure there are no execution privileges on upload targets;
- Do not pass user supplied data in web redirects;
- Application and resource files must be read-only.

Authentication and password management:

- Use a centralized implementation for all authentication controls;
- Use standard, tested authentication services whenever possible;
- Require authentication for all pages and resources except those intentionally public;
- Authentication failure notification should never give hints about which of the provided credentials is wrong;
- Change all vendor-supplied default passwords and user ids;
- Ensure passwords have security policies (eg. Minimum length, non-alphanumerical characters, uppercase & lowercase…);
- Password entry should be obscured on user interface;
- Password reset should send email only to pre-verified email addresses and support a link which can be used only once;
- Temporary passwords and links should have a short expiration time.

Session Management:

- Set domain and path for cookies involved in session management;
- Generate new session for each authentication;
- Generate new session when changing from HTTP to HTTPS;
- Application should use HTTPS protocol when possible;

- Do not expose session identifiers in URL, error messages and logs;
- Logout should be available only from pages protected by authorization;
- Protect server-side session data from unauthorized access.

Authorization and access control:

- Access controls should fail securely;
- Deny all access if application can't access to security configuration information;
- Enforce authorization request on every request including AJAX and Flash originating ones;
- Avoid whenever possible to save data on client-side;
- Restrict access to protected URLs only to authorized users;
- Restrict access to protected functions only to authorized users;
- Restrict access to services only to authorized users;
- Restrict access to application data only to authorized users;
- Restrict access to user and data attributes only to authorized users;
- Restrict access to access policies only to authorized users.

Data Protection:

- All cryptographic functions used to protect sensitive information must be implemented server-side;
- Application keys must not be embedded in an application;
- Restrict users only to those functionalities, data, system information that are required to perform relevant tasks;
- Protect server-side source code from being downloaded;
- Do not store passwords, connection strings or other sensitive information in clear text or in any non – cryptographically secure manner in client-side;
- Remove comments in production code;
- Remove unnecessary application and system documentation;
- Do not include sensitive information in HTTP GET requests.

Error handling and logging:

- Do not disclose sensitive information in error responses;
- Use error handlers which do not display debug information or trace stacks;
- Implement generic error messages and use custom error pages;
- Restrict access to log information to authorized individuals;
- Use a master routine for all logging operations;
- Do not store sensitive information in log files;
- Application should manage errors and not rely on server configuration.

System configuration:

- Ensure servers, frameworks and system components are running the latest approved version;
- Ensure servers, frameworks and system components have all patches issued for the version in use;
- Turn off directory listings;
- Remove all unnecessary files and functionality;
- Remove test code and functionality not intended for production use prior to deployments;

- Disable unnecessary HTTP methods;
- Prevent disclosure of directory structure (eg. Robots.txt);
- Isolate development and quality assurance environments from one.

Environments separation:

- Testing and staging environments are separated from the production environment.
- No actual customer data is used in the development or test environments.

Confidentiality agreement:

- All new hires are screened through the hiring process and required to sign Non-Disclosure and Confidentiality agreements.

# 2 Content security

THRON is a software service designed to organize, manage and deliver companies' digital content in a secure and controlled way.
The best worldwide delivery performance is often achieved by delivering static assets, preventing application logic to be involved in content delivery chain; enterprise level security, on the other hand, needs application level controls in order to authorize users and accesses to resources.
In order to fully understand how delivery performance and security are achieved, different content actions have to be considered separately.

## 2.1 Encryption (in transit)

All content and communications can be performed by using HTTPS protocol.

## 2.2 Encryption (at rest)

All content files are encrypted when stored by using SSL256.

## 2.3 Content "playback" from THRON's console

THRON is a complete solution which allows to distribute content both over the Internet or inside a corporate Intranet, both delivery types leverage a powerful acceleration system (CDN for Internet distribution, THRON Multimedia Proxy for Intranet distribution) to grant best performance and accessibility to users providing both caching and "collapsed forwarding" techniques to content delivery.
Caching is a mandatory technique to leverage geographical distribution: content is stored on several nodes to grant high availability, but such nodes won't grant sufficient network proximity to provide a fast and reactive response to user requests, this is why a CDN approach is used: cloud nodes act as origin servers while different, distributed, smaller nodes act as edge servers.
Edge servers act as "proxies": they store and delivery content to user requests when those requests are valid.
There are two different class of content when playback is involved:

- Content which can be played/viewed by web browser (we will refer them as multimedia-type or document-type)
- Generic files

Multimedia-type and document-type:

- Video, Audio: they can be used inside browser by leveraging embedded browsers multimedia capabilities (HTML5).
    - Video and Audio content can be rendered in different ways depending on device, operating system and configuration: streaming or (progressive) download. When using streaming, content is fragmented in different smaller files (chunks) and in such case content is identified by a descriptor: a text file which acts as an "index" for all chunks composing the content itself.
    - THRON protects the descriptor delivery from unauthorized accesses.
- Image content cannot be delivered by using streaming because information is small and can be kept inside a single digital file. Images do not have descriptors.
- Document-type content originates from MS Office, PDF and EPS files: they can be accessed inside browser by using an integrated HTML5 player.

Generic file content:

- Generic file content is downloaded using a session protected web service when accessed from THRON Dashboard. Since there is no embedded way for the browser to render this type of content, content must be downloaded from users and opened using their favourite players.

## 2.4 Content "playback" outside THRON's console: Shareboard and content sharing

Shareboard is the console section where users can assess and update where content is available and who has access to it.
THRON session cannot be used to authorize access when embedding a content on a personal website, social network or sending a content by email; this is why a "shared key" mechanism is used: when creating a new "share" from Shareboard a "share token" is created by THRON. Share token acts as key enabling THRON player for content playback/access.

Removing a share relation from Shareboard will deactivate token, making all access to content that used that key invalid.

### 2.5 Original content (source file)

THRON dashboard allows users who have proper access rights to download original file; this action is meaningful for multimedia-type and document-type content since there is a difference between data used for content playback and original file provided by authors (multimedia and documents content are transcoded to create different formats and qualities).
On generic-file content there is no difference between the original content (file sent by the user) and data used for playback.
Original content download is available only within THRON dashboard as an authenticated service only.

### 2.6 Content storage

Content is always stored on more than one cloud node at a given time, each cloud node is provided on different geographical positions and buildings to ensure content will not be lost in case of physical catastrophe on a specific region.
Each Cloud node includes a full content storage which is not accessible from the Internet: content storage may be accessed internally by THRON components only.
Content, in order to be delivered, is served by a file server which operates on authenticated sessions; any application level operation (read, write, update) on content needs to be originated from an authenticated session (this is also needed to grant ACL compliance) matching the user who is accessing content.
An internal directory system is used to manage user authentication and authorization.

## 3 User security

User information, group structure and access rights are stored on databases. Different databases are used for specific purposes: very big data sets which need to provide fast read access but have few write requests are stored on no-Sql databases;
Databases are not directly exposed to the Internet: databases are located on the last tier of architecture and can be accessed by technicians using a point-to-point VPN to access specific management vlans. Only application servers can access databases by using authenticated and authorized sessions.
A different authorization level is required to access user data in a read-only mode compared to read-write mode.

### 3.1 User password

Passwords are never stored as clear text on databases or any other THRON component, passwords hash are stored and a hash&compare mechanism is applied to authorize users.
Consequence to this security design choice is that there is no way to restore a lost password: passwords can only be replaced by administration-level users.

# 4 Privacy and user data management

THRON values user privacy and has always been working with the interest of its users as an objective, so it is just natural to comply with European General Data Protection Regulation and, in general, safeguarding the use of data just for clear, declared, purposes.
As a proof of this commitment, THRON's contract places THRON's customer as the Data Controller (thus owning all is contacts data) and THRON will act as Data Processor for the scope of delivering the DAM with Content Intelligence service.
THRON collects Personal Data only for users that need to access THRON console, this is because of the need to identify them for management purposes.
THRON does not automatically collect personal data regarding contacts (users and devices that access content), unless the Data Controller (THRON's customer) acquires user consent and transmits data to THRON for processing.
Check details on the extended privacy notice:

https://<your clientId>-cdn.thron.com/shared/assets/privacy/extendedprivacy.html

### 4.1 Collected data

THRON users accessing the console are required to provide name, surname, email, company and an optional profile picture.
THRON contacts do not require any data to be provided. After processing geographical lookup at country/region/city level, IP Address is masked so that it's not stored in THRON's systems preventing any accurate identification of the source of the events.

# 5 Cookie Policy

Cookies consist of portions of code installed in the browser that assist the Owner in providing the service according to the purposes described. Some of the purposes for which the Cookies are installed may also require the User's consent.
Details on which cookies are used by THRON can be found in the extended privacy notice:

https://<yourClientId>-cdn.thron.com/shared/assets/privacy/extendedprivacy.html

## 5.1 Technical Cookies and Cookies serving for aggregated statistical purposes

### 5.1.1 Technical purposes that require cookies for service delivery

THRON uses Cookies to save the User's session and to carry out other activities that are strictly necessary for the operation of the same, for example in relation to the distribution of traffic.

### 5.1.2 Other purposes that require cookies

THRON uses Cookies to save browsing preferences and to optimize the User's browsing experience. Among these Cookies are, for example, those to set the language and the currency or for the management of first party statistics employed directly by the Owner of the site.

## 5.2 Other types of Cookies or third-party tools that might use them

Some of the services listed below collect statistics in aggregated form and may not require the consent of the User or may be managed directly by the Owner - depending on how they are described - without the help of third parties.
If any third party operated services are listed among the tools below, these may be used to track Users' browsing habits – in addition to the information specified herein and without the Owner's knowledge. Please refer to the privacy policy of the listed services for detailed information.

### 5.2.1 Analytics

The services contained in this section enable the Owner to monitor and analyze web traffic and can be used to keep track of User behavior.

### 5.2.2 Interactions with external social networks and platforms

This type of service allows interaction with social networks or other external platforms directly from the pages of THRON. The interaction and information obtained through THRON are always subject to the User's privacy settings for each social network. This type of service might still collect traffic data for the pages where the service is installed, even when Users do not use it.

#### Facebook Like button and social widgets (Facebook, Inc.)

The Facebook Like button and social widgets are services allowing interaction with the Facebook social network provided by Facebook, Inc.
Personal Data collected: Cookies and Usage Data.
Place of processing: US – **Privacy Policy**

#### Twitter Tweet button and social widgets (Twitter, Inc.)

The Twitter Tweet button and social widgets are services allowing interaction with the Twitter social network provided by Twitter, Inc.
Personal Data collected: Cookies and Usage Data.
Place of processing: US – **Privacy Policy**

#### LinkedIn button and social widgets (LinkedIn Corporation)

The LinkedIn button and social widgets are services allowing interaction with the LinkedIn social network provided by LinkedIn Corporation.
Personal Data collected: Cookies and Usage Data.
Place of processing: US – **Privacy Policy**

#### Google+ +1 button and social widgets (Google Inc.)

The Google+ +1 button and social widgets are services allowing interaction with the Google+ social network provided by Google Inc.
Personal Data collected: Cookies and Usage Data.
Place of processing: US – **Privacy Policy**

## 5.3 How can I manage the installation of Cookies?

In addition to what is specified in this document, the User can manage preferences for Cookies directly from within their own browser and prevent – for example – third parties from installing them. Through the browser preferences, it is also possible to delete Cookies installed in the past, including the Cookies that might possibly have saved the consent for the installation of Cookies by this website. It is important to note that by disabling

all Cookies, the functioning of this site may be compromised. Users can find information about how to manage Cookies in their browser at the following addresses: **Google Chrome**, **Mozilla Firefox**, **Apple Safari**and **Microsoft Windows Explorer**.
In the case of services provided by third parties, Users can exercise their right to withdraw from the tracking activity by utilizing the information provided in the third party's privacy policy, by clicking the opt-out link – if provided – or by contacting the third party.
Notwithstanding the above, the Owner informs that Users may take advantage of: **Your Online Choices**. This service allows Users to select their tracking preferences for most of the advertising tools. The Owner thus recommends that Users make use of this resource in addition to the information provided in this document.

### 5.4 Data Controller and Data Processor

Such roles are defined in THRON Terms and Conditions, you can review them online at:
https://www.thron.com/en/general-terms-and-conditions

# 6 Web Services/API security

THRON is often referred as a platform because every user/content/configuration information can be accessed and updated by using Web Services instead of web console. There are two different Web Service types:

- Regular Web Services
- High request rate Web Services

### 6.1 Regular Web Services

This distinction is relevant to security because Regular Web Services are used for all "state changing" actions (like update and delete, reset user password) and for all "secure" actions (like download original content or get user profile data). Regular Web Services need an authenticated session in order to be invoked and to use an authorized user to perform requested actions.

### 6.2 High Request Rate Web Services

HRR web services are used when providing worldwide scale, high invoke frequency services used for content delivery (e.g., get content descriptor or get content details) or used by embedded code, which may be used on

unpredictable scale websites. HRR web services do not require authenticated sessions, they will only process public data and do not provide an authorization mechanism: when content are being published on external websites or social networks they became public and must be accessible from any unauthorized user.

## 6.3 Web Services used by external applications

THRON provides two different ways to integrate external applications:

- Custom applications: created using dashboard, allows to create/integrate applications to use /create content;
- Integration to 3rd party applications: created by administration level users, allows to integrate 3rd party systems to sync users database or access control lists;

Custom applications are intended for developers who need to use THRON's content to create engaging web or mobile applications exploiting existing metadata structure, content transcoding and content organization (DAM-like).
Enterprises usually provide their own users and groups structure (often on a LDAP system) and need to integrate such structure in THRON: in order to accomplish such integration tasks (which may also involve content, access rights or any other data type or feature set) a complete set of administration web services is available.
Custom Applications
Each custom application is associated to an application user. All application actions will be invoked by such user, disabling or removing the user will prevent the application to access information
within THRON. Application user will be provided access rights depending on tasks the application is meant to achieve: if custom application will just show a content carousel, a read only access will be provided; if custom application will generate user content (like an UGC contest), a read-write access will be provided for specified folders.
Custom applications will always be executed inside a well-defined sandbox, they can access user or access rights data as read-only.
Integration services
Enterprise deployments require a tighter integration with existing systems and infrastructure, this usually involves users data, groups structure and access rights mapping between users and groups to content or functions.
Although all those information can be adjusted/configured by using Web Services, those integrations do not follow the same path as custom applications to ensure a better security approach.
When designing an enterprise integration, THRON's dedicated team will identify which administration-level web services are needed and will create a specific user (or group of users) which will be able to use only the defined web services.
A source IP based protection can also be used to limit where those services can be accessed from.

# 7 Architecture

THRON is a modern software application based on a modern cloud computing architecture, there are several "nodes" which are fully independent: each one of them contains all information and functionalities needed to deliver complete service to users.
Each node is synchronized to other nodes by THRON core services, this architecture grants high availability and allows content and data to be redundant to cope with any accidental failure (hardware or physical).
Internet delivery is accelerated by a CDN service to give best performance to users regardless of their physical distribution.

## 7.1 High level architecture

The following diagram shows a logic-level architecture

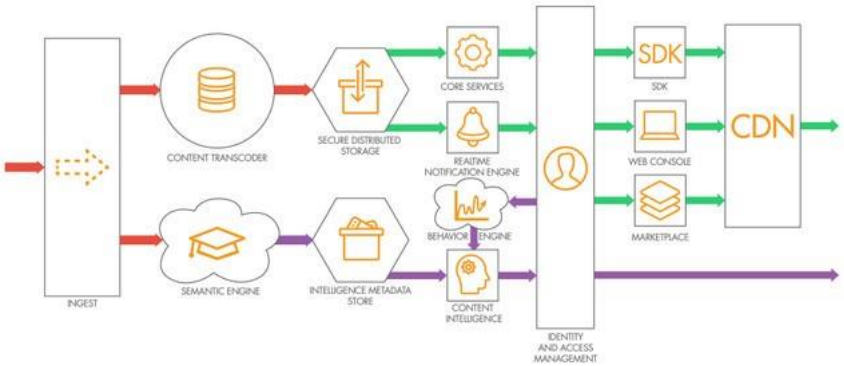


*Figure 1: red arrows for content ingestion flow, green arrows for content delivery flow, purple arrows for intelligence flow*

Being a cloud computing architecture each component shown before is actually composed by several virtual instances, whose amount can freely change during the day based on load requests. Each component "lives" inside a dedicated VLAN with specific access control definitions to rule inbound and outbound communication (even inside the same node).
Each node's application and data is protected by a firewall system implementing intrusion detection mechanisms.
There are just 3 available protocols to access THRON: HTTP, HTTPS and FTP; FTP and HTTP can be used to push content to platform, while the only protocol available to access Dashboard and content is HTTPS.
A software load balancer distributes traffic across backend services accordingly to custom rules, at the same time it offers SSL Offloading to boost internal node communications.

## 7.2 Uptime and availability

- Uptime: THRON maintains a publicly available system-status webpage that includes system availability details, scheduled maintenance, service incident history, and relevant security events.
- Redundancy: THRON's cloud architecture offers service clustering and network redundancies to eliminate single point of failure. Our backup policies ensure customer data is actively replicated across all systems and facilities.
- Disaster Recovery: Our disaster recovery program ensures that our services remain available or are easily recoverable in the case of a disaster. This is accomplished through building a robust technical environment, creating disaster recovery plans, and testing.

## 7.3 Service-based SLA

THRON's storage has a reliability of 99.9999999% thanks to its geographical distribution.
Some components might require some time for the propagation of their configurations (for example geographic scale DNS), for this reason the stated SLA, corresponding to the minimum between all of the components' SLA, is of 99.9%, never exceeded so far.
The SLA related to assets delivery is 100%.
The following control procedures are adopted:

- Monitoring of all components from geographically distributed sources
- Real user monitoring (performance and geographic availability)
- Application server monitoring (service delivery performance)
- Internal network monitoring (connection between the components)
- Web services monitoring (availability and integrity of web services)

# 8 Datacenters

Cloud computing infrastructure is based on physical datacenters located in different world regions.
Each structure is designed to grant maximum security parameters available today, both physical and network access are strictly regulated.
All systems, networked devices, and circuits are constantly monitored by both THRON and the co-location providers.
All datacenters follow the strictest regulations and achieved the following certifications:

| | | |
|---|---|---|
| SOC 1/SSAE 16 - ISAE 3402 / SOC 2 | SOC 3 | ISO 27001 |

Trained datacenter personnel protect physical access to infrastructure; strict surveillance measures are active to ensure only authorized maintenance technicians can access servers and network devices spaces.

### 8.1 Fault management

All monitoring systems are connected to automatic alerts that are managed by a presidium 24/7.
All systems are deployed in clusters, there is an automatic failover management: hardware failures (disk, memory) occur, especially given the number of servers in use, and they are handled automatically with no service disruption.
Platform upgrades are performed with no service disruption aswell; if a downtime is expected before an extraordinary update or a maintenance intervention, all customers will be informed in advance. As of real-time stateful protocols (such as web socket), the updates will trigger an automatic reconnection.

## 9 Conclusions

THRON is a software service designed to organize, manage, and deliver companies' digital content in a secured and controlled way.
It allows companies to map their organization using simple yet powerful users and groups concepts as well as define access control rights for all content; content access restrictions applies both to real users and external applications or websites, allowing company to always know which content are used, where they are used, who is using them and when he's using them.
Sensible or business critical information may be stored and managed by THRON, this is why security has been considered from the design phase and managed though the development process in order to minimize security flaws and be compliant with modern requirements.
Being flexible and providing web services access to external users or applications, an enterprise level security approach is provided as part of the service: to be able to sandbox all applications and precisely define which services/actions may be performed even for administration-level users.