Description

ThreatManager is a system that allows government staff to track and manage information and communications technology (ICT) threats across all internal systems (Government of the Netherlands, n.d.). Pillai's (2017) three key aspects of information security are relevant to this system:

- Confidentiality is important as details of the public and members of staff will be stored. Leaking of data is not only a GDPR issue but could cause reputational damage to this system itself and the Dutch National Police in general.
- Integrity is important because any tampering with the information being stored could result in public threats not being investigated or resolved.
- Availability: the system cannot be inaccessible for any significant length of time as this will delay the reporting of threats.

Operating Environment

- Client-server architecture (for a justification please see Open University (N.D)).
- Linux
- Python programming language.
- Libraries: pip, python-dotenv, Flask, flask-wtf, flask-sqlalchemy, flask-migrate, flask-login, email-validator, Flask-email, pyjwt, Bootstrap, pyotp, pytest, pytest-cov, pylint, flake8, pyflakes, safety, pandas, LGTM

Non-functional requirements

Normalization

The main objective is to create a well-structured relational database that fulfils the general definition of the Third Normal Form. This will be achieved by assessing normalisation whilst modelling the database through an Entity Relationship Diagram (ERD) as advised by Connolly and Beg (2015).

Software quality attributes

Pillai's (2017) key aspects to creating modifiable and flexible systems will be followed:

- Readability: Code will be clear and concise. Comments will be made throughout the code and README files will document the workings of the system.
- Modularity: The system will be divided into well-encapsulated modules.
- Reusability: The DRY principle will be followed.
- Maintainability: The system will be easy to update and maintain.

Architecture patterns

The system will make use of the Model View Controller (MVC) pattern to ensure good system design (Pillai, 2017). Furthermore, an API that provides a distributed service interface will follow a Representational State Transfer (REST) architecture style (Richards, 2006).

Security

ThreatManager will maintain a database of all the flaws within internal and related government systems, therefore security will be of utmost importance. Eight out of the ten of

OWASP’s common vulnerabilities have been identified below as relevant to ThreatManager (OWASP, N.D.).

A1:2017-Injection:

- By using an ORM library (flask-sqlalchemy) to access the database, the application will rely on parameterized statements (SQLAlchemy, N.D.).

A2:2017-Broken Authentication:

- Regex will be used to do weak-password checks.
- Multi-factor authentication in the form of Time-based One-Time Password (TOPT).
- A maximum of 5 failed login attempts per hour.
- Once a user successfully logs in, a random session ID will be generated that will be used to verify the user before being destroyed upon logout.

A3:2017-Sensitive Data Exposure:

- Encryption of personal data in the database.
- HTTPS protocol to be used (provided by NCSC).
- Automatic deactivation of staff accounts if not accessed after 12 months.

A5:2017-Broken Access Control:

- JWT tokens used for authentication with an expiration time of 2 hours.
- Rate limit of 10 requests per hour on the API.
- Principle of least privilege (OWASP, N.D.).
- Role-based access control (Auth0, N.D.).

A6:2017-Security Misconfiguration:

- Proper error handling to not expose sensitive information.
- Up to date, tested libraries and code on the main application.
- Streamlined to not include unnecessary code or documentation.

A7:2017-Cross-Site Scripting XSS:

- Flask, (N.D.) framework comes with security features to mitigate XSS such as:
    - Flask as a framework configures Jinja2 to automatically escape all values.
    - Explicitly defining content-type for content uploads.
    - Proper quoting of attributes when using Jinja2 expressions.

A9:2017-Using Components with Known Vulnerabilities:

- The safety library will check for vulnerabilities in the project dependencies.

A10:2017-Insufficient Logging & Monitoring:

- Both errors and user actions will be logged and stored in a database.

Business requirements

Please see the activity diagram in the Appendix A.

Assumptions

- This project serves as a minimum viable product (MVP). As such, not all features are discussed or covered.
- The hosting environment will be provided by the National Cyber Security Centre.
- Dutch Law implements GDPR legislation in a similar way to the British legal system.
- The hosting environment meets the minimum specification required.

Data Protection

The only personal data stored will be: email address, name and surname

GDPR

All the below is based on guidance from the Information Commissioner’s Office (ICO, 2021).

Proving technological security measures have been taken:

- Outlined in the OWASP table above.

Storage limitation/retention period:

- User data will be kept for no longer than six months after resolution of an issue.

Other issues:

- Lawful basis of processing is legitimate interest (to help improve the link between the public reporting security issues and government organisations responding).
- Right of erasure will be achieved through requests from members of the public being passed to developers.
- Notification of data breaches to the Dutch equivalent of the ICO will be within 72 hours. This will be implemented through the daily review of system logs, with a policy of immediately reporting data leakage to the NCSC Data Protection Officer.
- No special categories of data will be processed (for example biometric data)

Testing plan

Pillai (2017) suggest that there are several aspects of practical software testing, such as functional testing, performance testing, usability testing etc. The below will be used with ThreatManager:

Black-box testing:

- Functional testing: adding, approving and editing cases across the various authenticated user roles will be tested by each member of the team.
- Compatibility testing: the system will be tested on various browsers.
- Usability testing: to test for adequate usability, several people outside of the team will be asked to test the software and their reviews taken onboard by the team.
- Acceptance testing: will be used to mark a task as done.
- Security testing: Verification of the authorization of different account roles, such as the specified role’s actions and sensitive data accessing, will be carried out throughout the development.

White-box testing:

- Unit testing for individual components will be conducted with at least 80% test coverage.
- Integration testing will be attempted.

The library *pytest* will be used for the unit and integration test, since it has rich inbuilt features and requires relatively less code, compared to other libraries like *unittest*. (Python Pool, 2021)

Several types of Linter will be used (as noted in the ‘operating environment’ section above). Experience-based testing and error-guessing will be used due to the limited time involved.

Code reviews

Code reviews will be done and at least one developer will have to accept or reject code.

References

Auth0. (N.D.) Role-Based Access Control. Available from: https://auth0.com/docs/authorization/rbac/ [Accessed 18 September 2021].

Connolly, T. & Beg, C. (2015) *Database Systems: A Practical Approach to Design, Implementation, and Management*. Global Edition. Edinburgh: Pearson Education Limited.

Flask. (N.D.) Security Considerations. Available from: https://flask.palletsprojects.com/en/2.0.x/security/ [Accessed 18 September 2021].

Government of the Netherlands. (N.D.) Fighting cybercrime in the Netherlands. Available from: https://www.government.nl/topics/cybercrime/fighting-cybercrime-in-the-netherlands [accessed on 1st September 2021].

Intellectual Property Office. (2014) Exceptions to copyright: Education and Teaching. Available from: https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/375951/Education_and_Teaching.pdf [Accessed 19 September 2021].

ISO. (2021) Guide to the General Data Protection Regulation (GDPR). Available from: https://ico.org.uk/media/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr-1-1.pdf [Accessed 16 September 2021)

OWASP. (N.D.) Access Control. Available from: https://owasp.org/www-community/Access_Control [Accessed 18 September 2021].

OWASP. (N.D.) Top 10 Web Application Security Risks. Available from: https://owasp.org/www-project-top-ten/ [Accessed 18 September 2021].

Pillai, A.B. (2017) *Software Architecture with Python*. Birmingham, UK: Packt Publishing Ltd.

Python Pool. (2021) Python Unittest Vs Pytest: Choose the Best. Available from: https://www.pythonpool.com/python-unittest-vs-pytest/ [Accessed 18 September 2021].

Richards, R. (2006) *Pro PHP XML and Web Services*. Berkeley, CA: Apress.

SQLAlchemy. (N.D.) Key Features of SQLAlchemy. Available from: https://www.sqlalchemy.org/features.html [Accessed 18 September 2021].

The Open University. (N.D.) An introduction to web applications architecture. Available from: https://www.open.edu/openlearn/science-maths-technology/introduction-web-applications-architecture/content-section-1.1 [Accessed 18 September 2021].

Appendices

Appendix A: Business requirements

Register

Preconditions	Users must have a valid email address (assigned by a government related organisation or partner) in order to have a *read* role that can be upgraded to *editor*, *approver* or *admin* role. Every other registration will be of *public* type.
Description	The user will register and receive a confirmation email that will require him/her to click on in order to complete the registration in addition to activating Time-based One-Time Password (TOTP) to complete two-factor authentication (2FA).

Login

Preconditions	Valid credentials.
Description	The user must authenticate before gaining access to the system. The user needs the following information: - Email address - Password - TOTP

Logout

Preconditions	The user must be logged in.
Description	The user logs out by pressing the corresponding button.

Create threat listing

Preconditions	The user must be assigned the public or editor role.
Description	Create a detailed threat listing and request for users with an *approver* role to reject, approve or resolve the threat. File uploads limited to PNGs, and JPEGs with a max file size of 5mb per file and up to 5 files can be attached to a listing.

View threat listing

Preconditions	A threat listing needs to exist.
Description	View a previously created threat listing. Can be viewed by any registered user in a *read-only* view and can be edited by a user with an *editor* role.

Delete threat listing

Preconditions	A threat listing needs to exist. The user must be assigned an *editor* role.
Description	A previously created threat listing can be deleted.

Reject threat listing

Preconditions	A threat needs to have been sent in for approval. The user must be assigned an approval role.
Description	A threat listing can be rejected if the information is insufficient.

Approve threat listing

Preconditions	A threat needs to have been sent in for approval. The user must be assigned an approver role.
Description	A threat listing can be approved and sent to the necessary security departments if sufficient information has been provided.

Resolve threat listing

Preconditions	A threat that was resolved. The user must be assigned an approver role.
Description	A threat listing can be marked as resolved if it was fixed.

Add comment

Preconditions	A threat listing needs to exist.
Description	A threat listing can have comments attached to it. A user with the public role can add a comment to the listing that was created by him/her. A user with an *editor* role can add a comment to any listing, while an *approver* can add a comment to only the listing that he/she rejected, approved or resolved.

Assign role

Preconditions	The user must be assigned an *admin* role. Roles need to exist in the database.
Description	A user with an admin role can *assign* roles to other users that do not already have an admin role assigned to them and that have requested an upgrade from their current role.

Authenticate through REST API

Preconditions	Valid client id, secret key and user credentials
Description	A secret key has to be assigned to a developer role user. Gaining access to other endpoints requires the user to be authenticated.

Download file through a REST API

Preconditions	The client must have a valid JWT token.
Description	A client with a valid JWT token and an api role, can access the API that will return a JSON response that will contain a temporary URL of the file to be downloaded.

[Please note: there are additional pages below]

Appendix B: Activity diagrams (please refer to attachment)

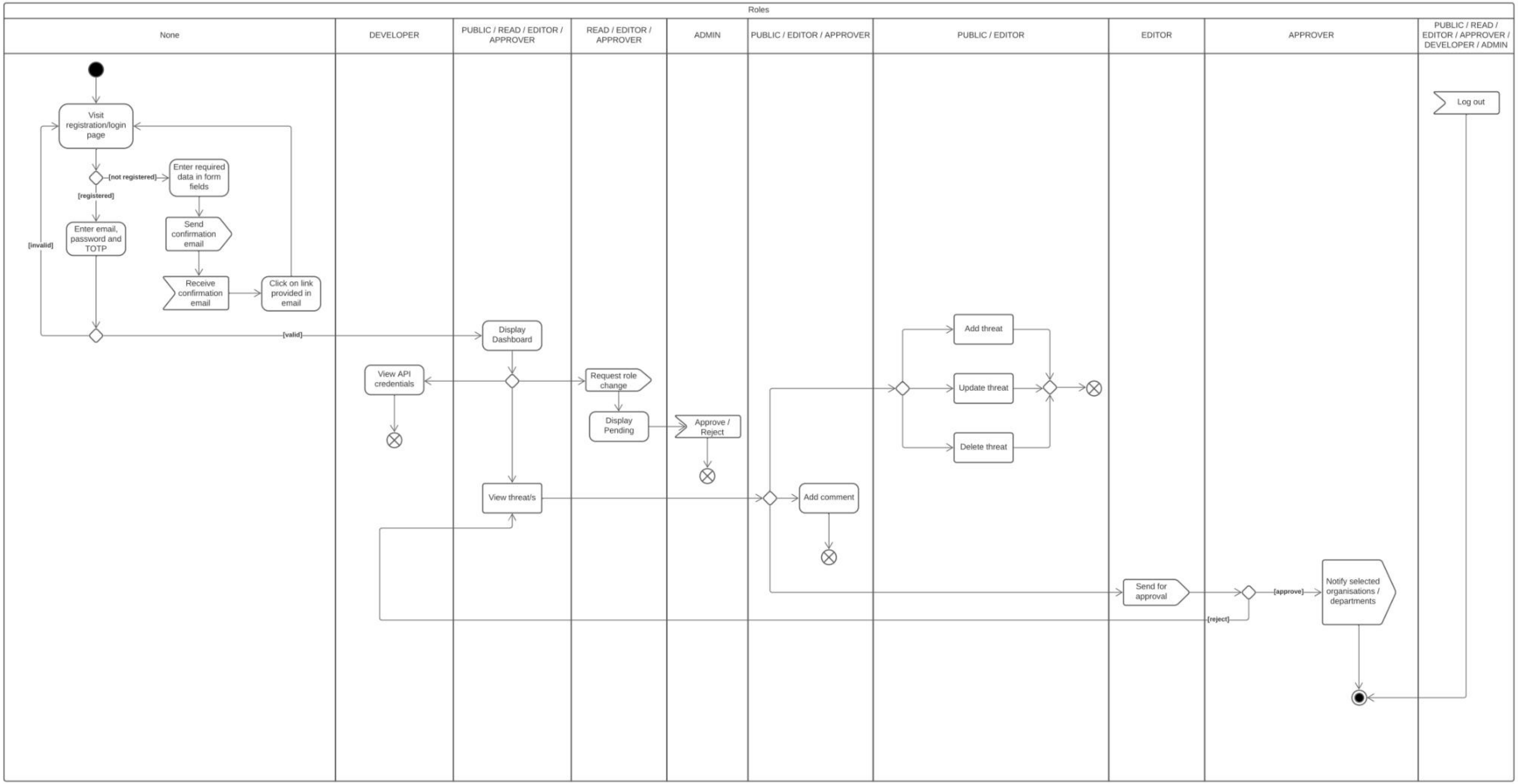


Figure 1: user’s activities of the monolith application

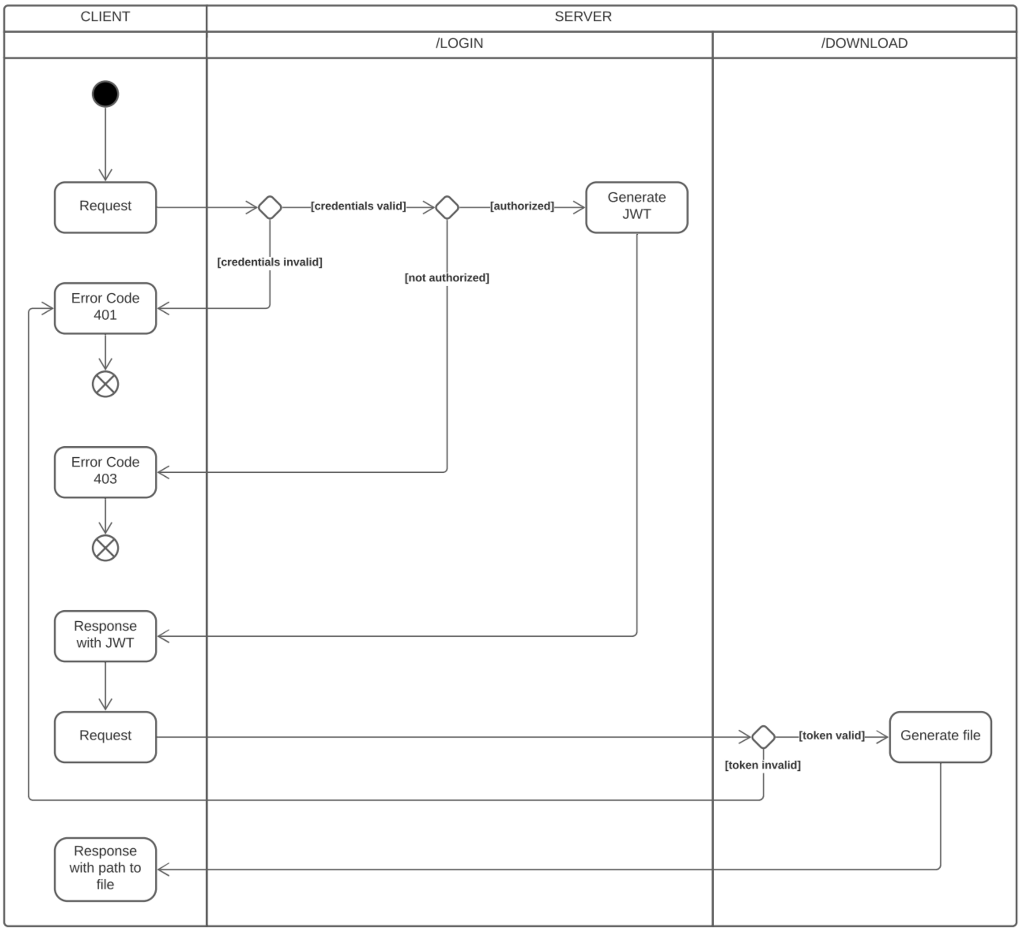


Figure 2: client’s activity of the API

Appendix C: Preliminary design of the application interface

Please note that the Dutch Police logo has been used under “exception to copyright” based upon educational exception guidelines by the Intellectual Property Office (2014). If this system is uploaded to the web (for example on an e-portfolio) then either a self-created logo or an image under a creative commons license will be used instead.

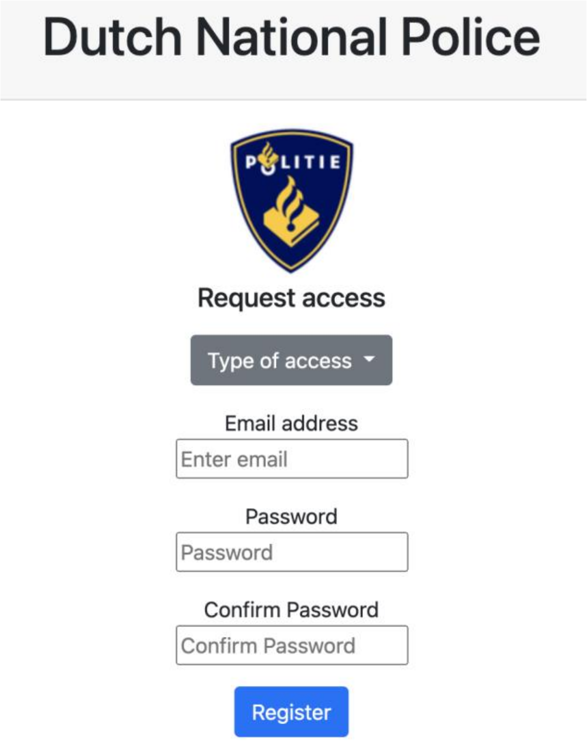


Figure 1: Registration

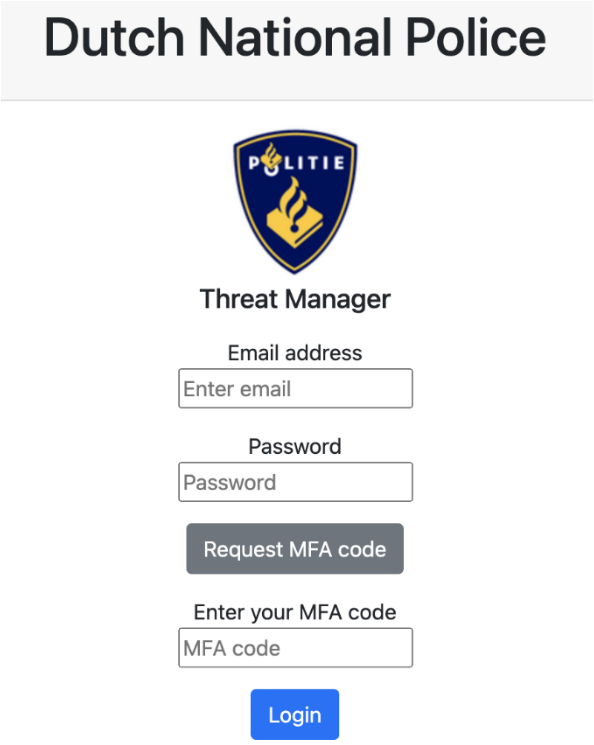
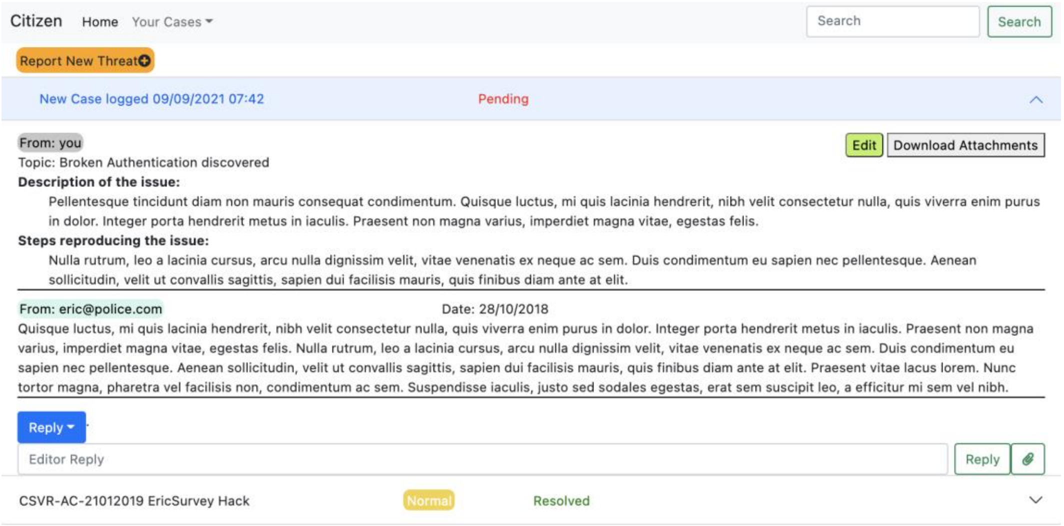


Figure 2: User login



New Case logged 09/09/2021 07:42 Pending

From: you

Topic: Broken Authentication discovered

Description of the issue:

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis.

Steps reproducing the issue:

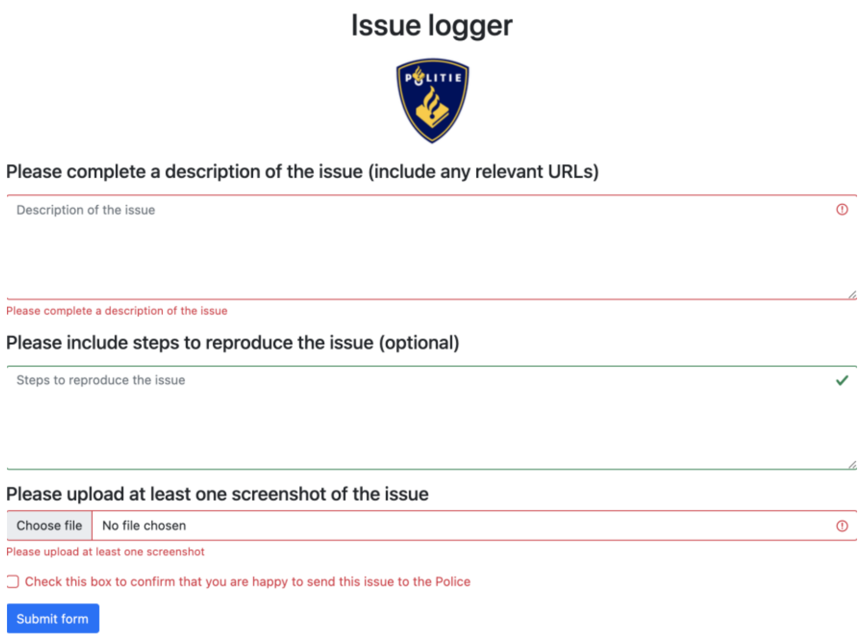
Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit.

From: eric@police.com Date: 28/10/2018

Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

CSVR-AC-21012019 EricSurvey Hack Normal Resolved

Figure 3: Citizen interface



Issue logger

Please complete a description of the issue (include any relevant URLs)

Description of the issue

Please complete a description of the issue

Please include steps to reproduce the issue (optional)

Steps to reproduce the issue ✓

Please upload at least one screenshot of the issue

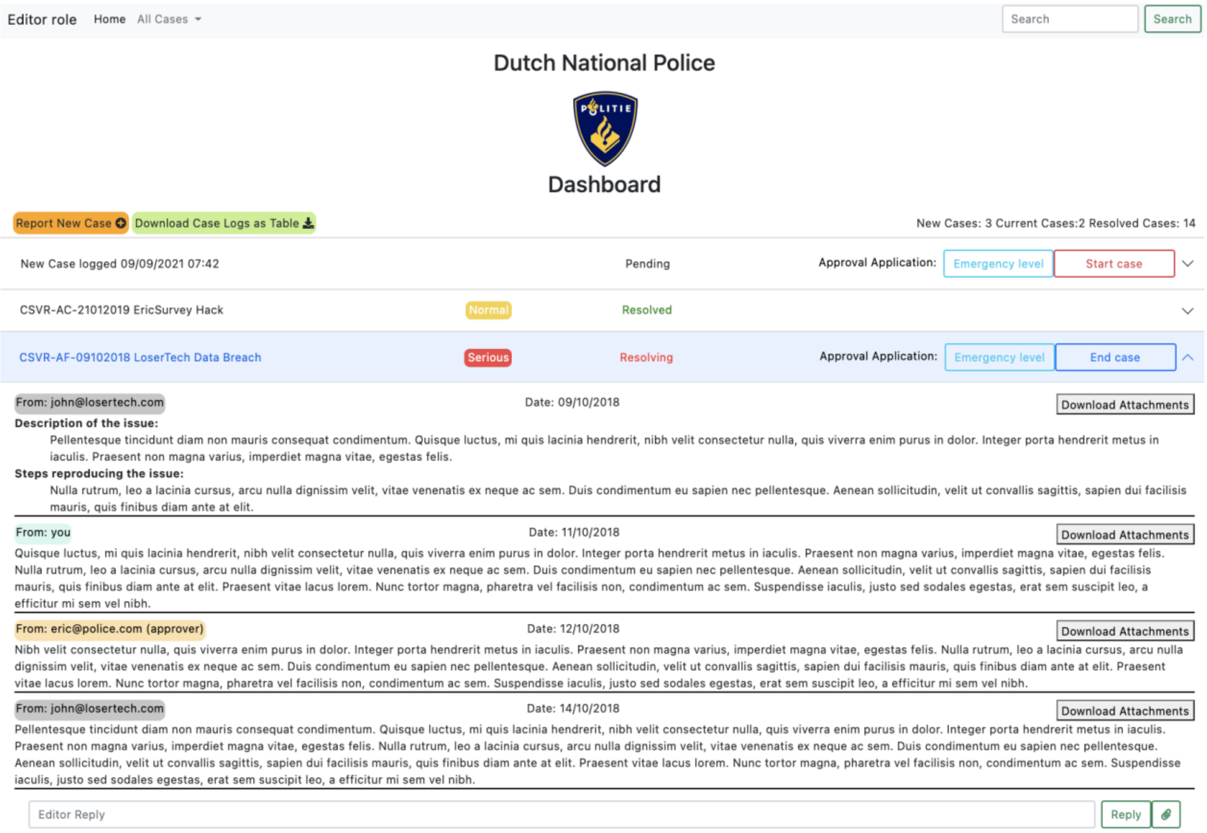
Choose file No file chosen

Please upload at least one screenshot

□ Check this box to confirm that you are happy to send this issue to the Police

Submit form

Figure 4: Form for reporting new issue / threat



Editor role Home All Cases Search Search

Dutch National Police

Dashboard

Report New Case Download Case Logs as Table

New Cases: 3 Current Cases:2 Resolved Cases: 14

New Case logged 09/09/2021 07:42		Pending	Approval Application: Emergency level Start case
CSVR-AC-21012019 EricSurvey Hack	Normal	Resolved	
CSVR-AF-09102018 LoserTech Data Breach	Serious	Resolving	Approval Application: Emergency level End case

From: john@losertech.com Date: 09/10/2018 Download Attachments

Description of the issue:

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis.

Steps reproducing the issue:

Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit.

From: you Date: 11/10/2018 Download Attachments

Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

From: eric@police.com (approver) Date: 12/10/2018 Download Attachments

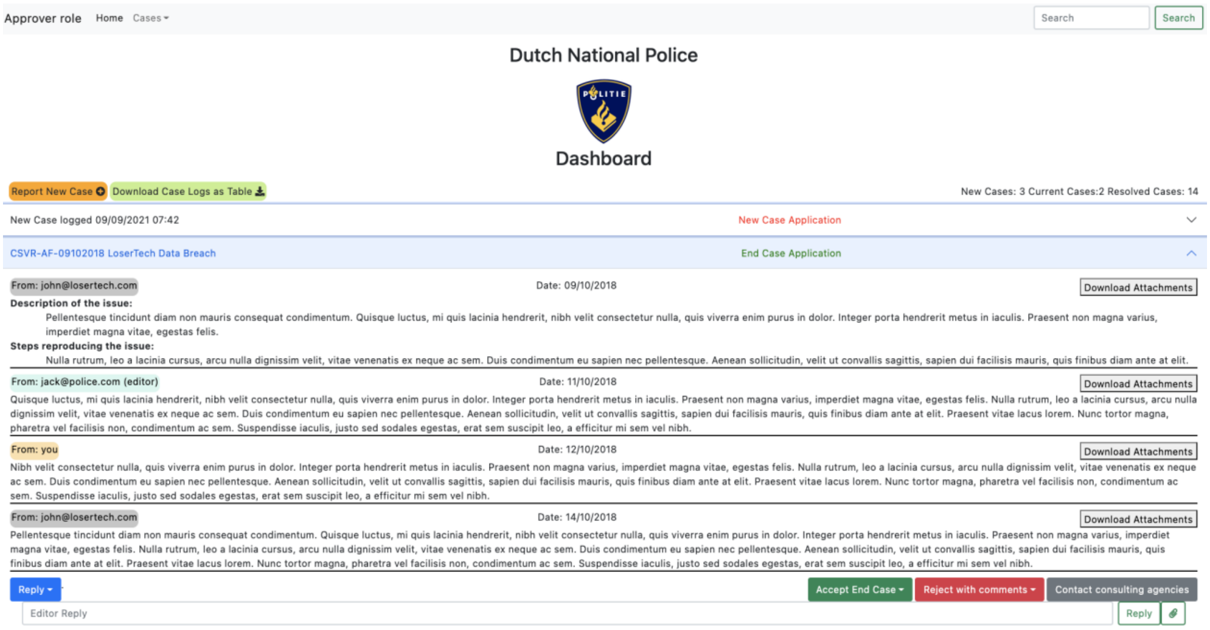
Nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

From: john@losertech.com Date: 14/10/2018 Download Attachments

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

Editor Reply Reply

Figure 5: police (editor) dashboard



New Case logged 09/09/2021 07:42 New Case Application

CSVR-AF-09102018 LoserTech Data Breach End Case Application

From: john@losertech.com Date: 09/10/2018

Description of the issue:

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis.

Steps reproducing the issue:

Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit.

From: jack@police.com (editor) Date: 11/10/2018

Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

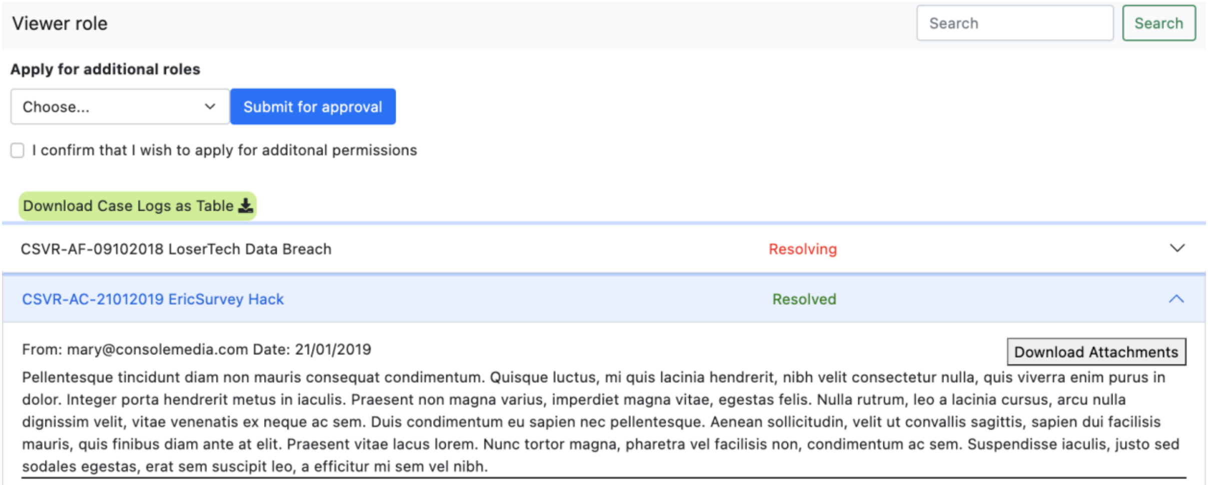
From: you Date: 12/10/2018

Nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

From: john@losertech.com Date: 14/10/2018

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

Figure 6: police (approver) dashboard


Viewer role
Search
Search
Apply for additional roles
Choose...
Submit for approval
□ I confirm that I wish to apply for additonal permissions
Download Case Logs as Table
Download Attachments

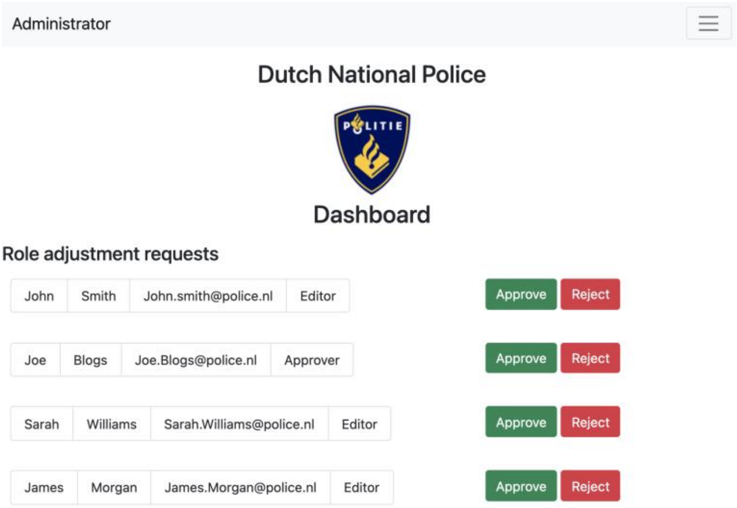
CSVR-AF-09102018 LoserTech Data Breach Resolving

CSVR-AC-21012019 EricSurvey Hack Resolved

From: mary@consolemedia.com Date: 21/01/2019

Pellentesque tincidunt diam non mauris consequat condimentum. Quisque luctus, mi quis lacinia hendrerit, nibh velit consectetur nulla, quis viverra enim purus in dolor. Integer porta hendrerit metus in iaculis. Praesent non magna varius, imperdiet magna vitae, egestas felis. Nulla rutrum, leo a lacinia cursus, arcu nulla dignissim velit, vitae venenatis ex neque ac sem. Duis condimentum eu sapien nec pellentesque. Aenean sollicitudin, velit ut convallis sagittis, sapien dui facilisis mauris, quis finibus diam ante at elit. Praesent vitae lacus lorem. Nunc tortor magna, pharetra vel facilisis non, condimentum ac sem. Suspendisse iaculis, justo sed sodales egestas, erat sem suscipit leo, a efficitur mi sem vel nibh.

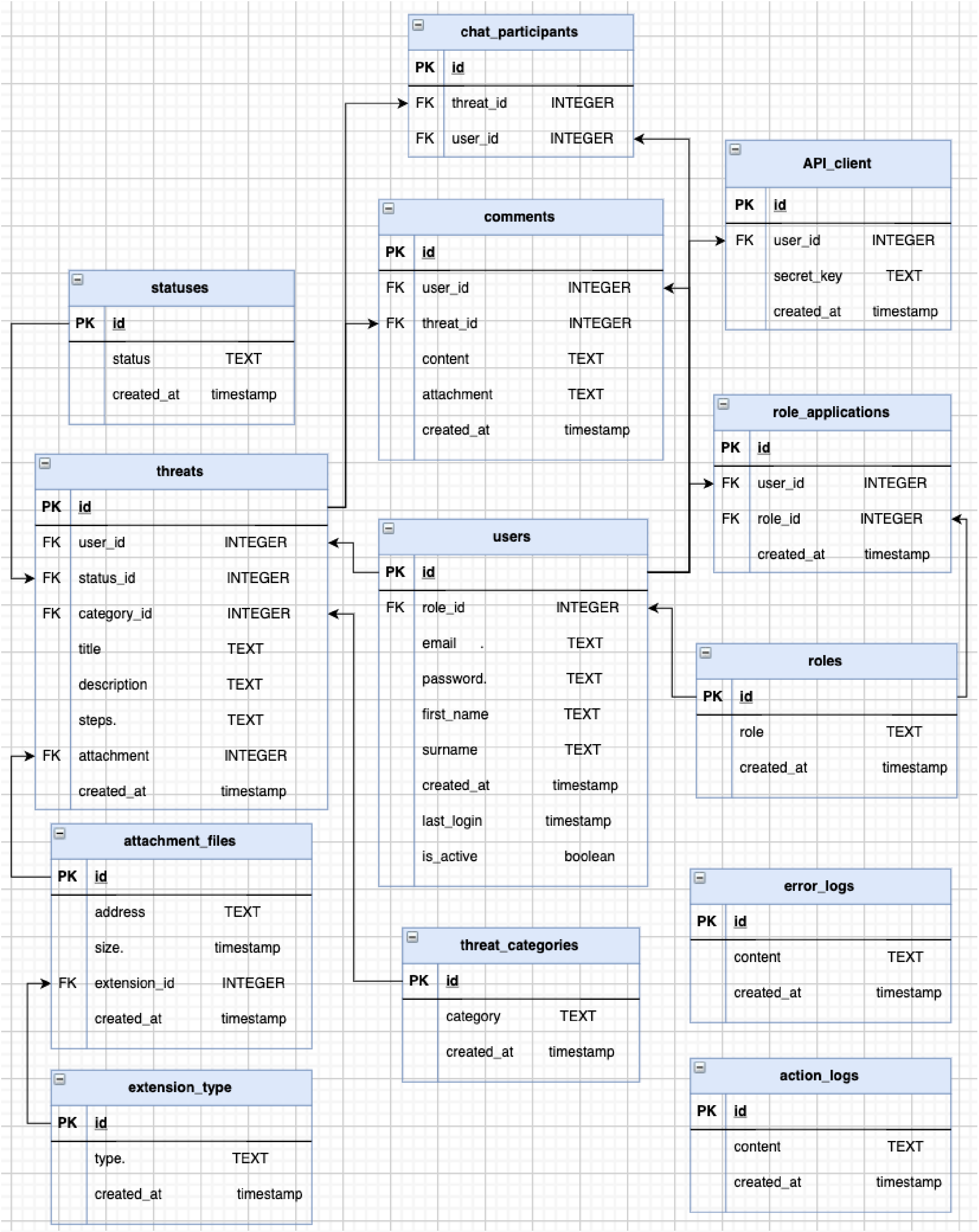
Figure 7: police (viewer) dashboard



John	Smith	John.smith@police.nl	Editor	Approve	Reject
Joe	Blogs	Joe.Blogs@police.nl	Approver	Approve	Reject
Sarah	Williams	Sarah.Williams@police.nl	Editor	Approve	Reject
James	Morgan	James.Morgan@police.nl	Editor	Approve	Reject

Figure 8: police (admin) dashboard

Appendix D: SQLite Database Diagram


chat_participants
PK id
FK threat_id INTEGER
FK user_id INTEGER
API_client
PK id
FK user_id INTEGER
secret_key TEXT
created_at timestamp
comments
PK id
FK user_id INTEGER
FK threat_id INTEGER
content TEXT
attachment TEXT
created_at timestamp
statuses
PK id
status TEXT
created_at timestamp
role_applications
PK id
FK user_id INTEGER
FK role_id INTEGER
created_at timestamp
threats
PK id
FK user_id INTEGER
FK status_id INTEGER
FK category_id INTEGER
title TEXT
description TEXT
steps. TEXT
FK attachment INTEGER
created_at timestamp
users
PK id
FK role_id INTEGER
email . TEXT
password. TEXT
first_name TEXT
surname TEXT
created_at timestamp
last_login timestamp
is_active boolean
roles
PK id
role TEXT
created_at timestamp
attachment_files
PK id
address TEXT
size. timestamp
FK extension_id INTEGER
created_at timestamp
error_logs
PK id
content TEXT
created_at timestamp
threat_categories
PK id
category TEXT
created_at timestamp
action_logs
PK id
content TEXT
created_at timestamp
extension_type
PK id
type. TEXT
created_at timestamp