The Hive Security Tool

Sumaiya Seaha, 1905033
Fariha Zaman Aurin, 1905049
Debjany Ghosh Aronno, 1905053

Department of CSE
Bangladesh University of Engineering & Technology

March 9, 2024

Table of contents

Table of contents

Unveiling TheHive: Your Security Ally

- Welcome to THE HIVE – Your Security Incident Response Wing!
- Just like GitHub brings developers together, TheHive unites security analysts in a collaborative heaven.
- Picture this: THE HIVE seamlessly integrates with powerhouse tools like CORTEX and MISP, automating the analysis of security incidents with unmatched precision and efficiency.

Why TheHive?

Rapid Detection, Collective Analysis: Swiftly identify and collaboratively dissect security incidents, ensuring efficient resolution of any security challenge.

Unveiling TheHive: Your Security Ally

- Welcome to THE HIVE – Your Security Incident Response Wing!
- Just like GitHub brings developers together, TheHive unites security analysts in a collaborative heaven.
- Picture this: THE HIVE seamlessly integrates with powerhouse tools like CORTEX and MISP, automating the analysis of security incidents with unmatched precision and efficiency.

Why TheHive?

Rapid Detection, Collective Analysis: Swiftly identify and collaboratively dissect security incidents, ensuring efficient resolution of any security challenge.

Unveiling TheHive: Your Security Ally

- Welcome to THE HIVE – Your Security Incident Response Wing!
- Just like GitHub brings developers together, TheHive unites security analysts in a collaborative heaven.
- Picture this: THE HIVE seamlessly integrates with powerhouse tools like CORTEX and MISP, automating the analysis of security incidents with unmatched precision and efficiency.

Why TheHive?

Rapid Detection, Collective Analysis: Swiftly identify and collaboratively dissect security incidents, ensuring efficient resolution of any security challenge.

Unveiling TheHive: Your Security Ally

- Welcome to THE HIVE – Your Security Incident Response Wing!
- Just like GitHub brings developers together, TheHive unites security analysts in a collaborative heaven.
- Picture this: THE HIVE seamlessly integrates with powerhouse tools like CORTEX and MISP, automating the analysis of security incidents with unmatched precision and efficiency.

Why TheHive?

Rapid Detection, Collective Analysis: Swiftly identify and collaboratively dissect security incidents, ensuring efficient resolution of any security challenge.

Table of contents

TheHive's Architecture Overview

1. **Frontend**
   - User interface where analysts interact.
   - Developed using AngularJS and Bootstrap.
   - Control panel for managing cases, tasks, and observables.
2. **Backend**
   - Core logic and heavy lifting.
   - Implemented in Scala, Akka, Play Framework, and Slick.
   - Ensures smooth processing of data and communication with the frontend.

TheHive's Architecture Overview

1. **Frontend**
   - User interface where analysts interact.
   - Developed using AngularJS and Bootstrap.
   - Control panel for managing cases, tasks, and observables.
2. **Backend**
   - Core logic and heavy lifting.
   - Implemented in Scala, Akka, Play Framework, and Slick.
   - Ensures smooth processing of data and communication with the frontend.

TheHive's Architecture Overview Contd.

3. Cortex

- Real-time analytics platform.
- Enhances intelligence with Scala, Akka, Play Framework, and Python.
- Analyzes data from the backend, adding an extra layer of insight.

4. Storage

- Essential memory bank for TheHive.
- Utilizes Elasticsearch, a distributed database.
- Stores all data securely for efficient retrieval.

Overview

Cortex is an integral component that enhances the capabilities of TheHive by providing real-time analytics and active response capabilities. It is used to process data from the backend, analyze observables, and perform actions on those observables, such as blocking an IP address or quarantining a file.

TheHive's Architecture Overview Contd.

3. Cortex

- Real-time analytics platform.
- Enhances intelligence with Scala, Akka, Play Framework, and Python.
- Analyzes data from the backend, adding an extra layer of insight.

4. Storage

- Essential memory bank for TheHive.
- Utilizes Elasticsearch, a distributed database.
- Stores all data securely for efficient retrieval.

Overview

Cortex is an integral component that enhances the capabilities of TheHive by providing real-time analytics and active response capabilities. It is used to process data from the backend, analyze observables, and perform actions on those observables, such as blocking an IP address or quarantining a file.

TheHive's Architecture Overview Contd.

3. Cortex

- Real-time analytics platform.
- Enhances intelligence with Scala, Akka, Play Framework, and Python.
- Analyzes data from the backend, adding an extra layer of insight.

4. Storage

- Essential memory bank for TheHive.
- Utilizes Elasticsearch, a distributed database.
- Stores all data securely for efficient retrieval.

Overview

Cortex is an integral component that enhances the capabilities of TheHive by providing real-time analytics and active response capabilities. It is used to process data from the backend, analyze observables, and perform actions on those observables, such as blocking an IP address or quarantining a file.

TheHive's Architecture - Continued

How It Works

The frontend serves as the user's window into TheHive, allowing for seamless interaction with cases and tasks. The backend, powered by Scala and Akka, processes the user's commands and orchestrates the entire operation.

Cortex, the analytical powerhouse, brings real-time insights to the table, using a combination of Scala, Akka, Play Framework, and Python. It acts like a smart assistant, enriching the analysis process.

The storage layer, backed by Elasticsearch, ensures that every piece of data is securely stored and easily retrievable. Think of it as TheHive's reliable memory bank, storing information for future reference.

TheHive's Architecture - Continued

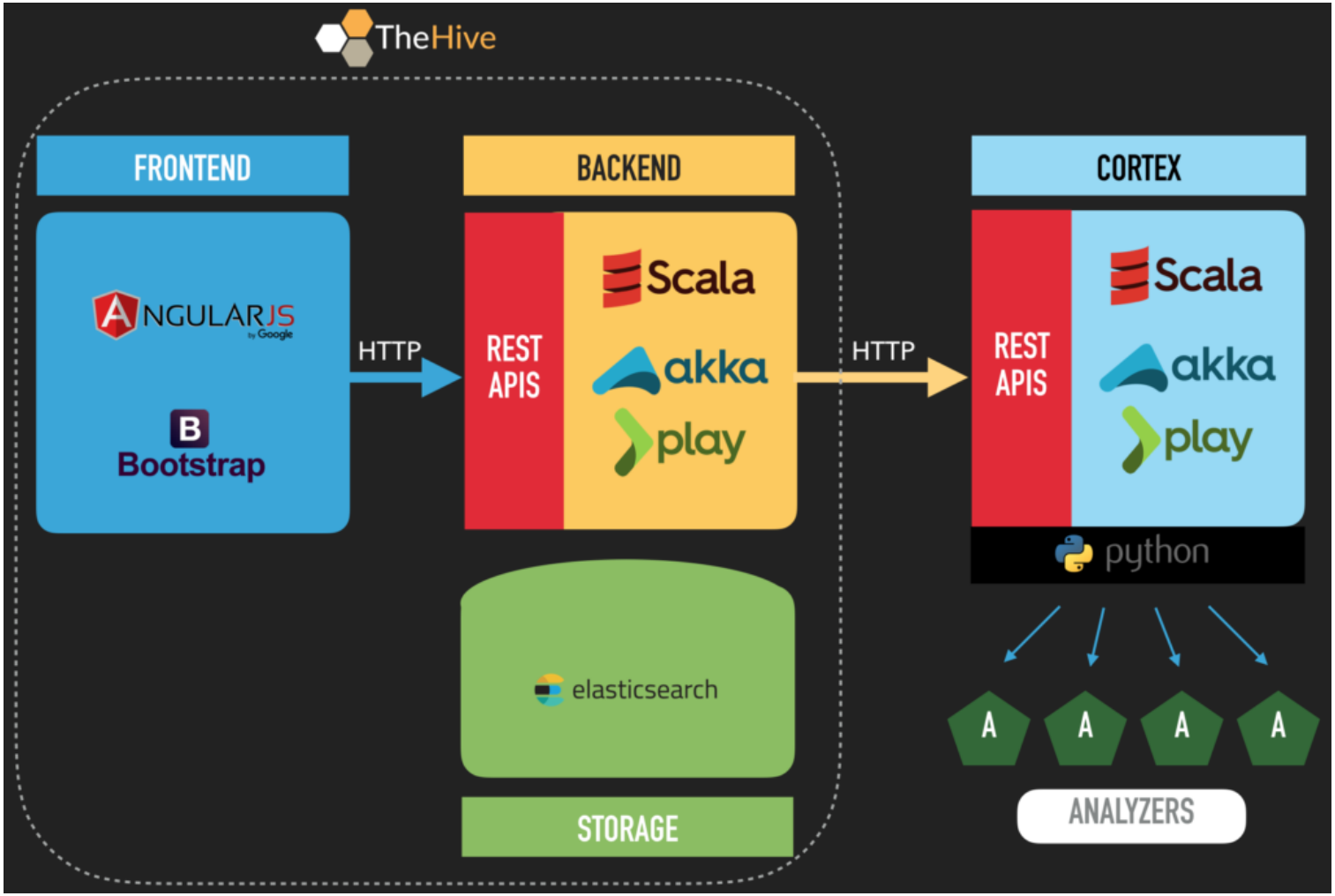


Figure: Overview of TheHive's Architecture

Table of contents

Key Concepts in TheHive

- **User:** TheHive supports two main user types – Admin and User.
- **Organization:** Admins can create organizations and manage users within them. Organizations group multiple users who collaborate on specific types of security incidents.
- **Case:** Users within an organization can create cases to document and manage incidents. Each case represents a specific security incident.
- **Task:** For each case, there can be one or more tasks aimed at resolving the incident. Tasks are similar to issues on platforms like GitHub and can be assigned to one or more users.

Key Concepts in TheHive

- **User:** TheHive supports two main user types – Admin and User.
- **Organization:** Admins can create organizations and manage users within them. Organizations group multiple users who collaborate on specific types of security incidents.
- **Case:** Users within an organization can create cases to document and manage incidents. Each case represents a specific security incident.
- **Task:** For each case, there can be one or more tasks aimed at resolving the incident. Tasks are similar to issues on platforms like GitHub and can be assigned to one or more users.

Key Concepts in TheHive

- **User:** TheHive supports two main user types – Admin and User.
- **Organization:** Admins can create organizations and manage users within them. Organizations group multiple users who collaborate on specific types of security incidents.
- **Case:** Users within an organization can create cases to document and manage incidents. Each case represents a specific security incident.
- **Task:** For each case, there can be one or more tasks aimed at resolving the incident. Tasks are similar to issues on platforms like GitHub and can be assigned to one or more users.

Key Concepts in TheHive

- **User:** TheHive supports two main user types – Admin and User.
- **Organization:** Admins can create organizations and manage users within them. Organizations group multiple users who collaborate on specific types of security incidents.
- **Case:** Users within an organization can create cases to document and manage incidents. Each case represents a specific security incident.
- **Task:** For each case, there can be one or more tasks aimed at resolving the incident. Tasks are similar to issues on platforms like GitHub and can be assigned to one or more users.

Key Concepts (Contd.)

- **Observables:** Data points representing relevant information in an investigation, such as IP addresses, domains, or hashes.
- **Analyzers and Responders:** Automation mechanisms for analyzing and responding to observables using external tools.
- **Alerts:** Notifications generated by external tools or manually created to flag potential security incidents.
- **Dashboards:** Overview pages providing insights into ongoing cases, tasks, and other metrics.

Key Concepts (Contd.)

- **Observables:** Data points representing relevant information in an investigation, such as IP addresses, domains, or hashes.
- **Analyzers and Responders:** Automation mechanisms for analyzing and responding to observables using external tools.
- **Alerts:** Notifications generated by external tools or manually created to flag potential security incidents.
- **Dashboards:** Overview pages providing insights into ongoing cases, tasks, and other metrics.

Key Concepts (Contd.)

- **Observables:** Data points representing relevant information in an investigation, such as IP addresses, domains, or hashes.
- **Analyzers and Responders:** Automation mechanisms for analyzing and responding to observables using external tools.
- **Alerts:** Notifications generated by external tools or manually created to flag potential security incidents.
- **Dashboards:** Overview pages providing insights into ongoing cases, tasks, and other metrics.

Key Concepts (Contd.)

- **Observables:** Data points representing relevant information in an investigation, such as IP addresses, domains, or hashes.
- **Analyzers and Responders:** Automation mechanisms for analyzing and responding to observables using external tools.
- **Alerts:** Notifications generated by external tools or manually created to flag potential security incidents.
- **Dashboards:** Overview pages providing insights into ongoing cases, tasks, and other metrics.

Table of contents

General Workflow of TheHive

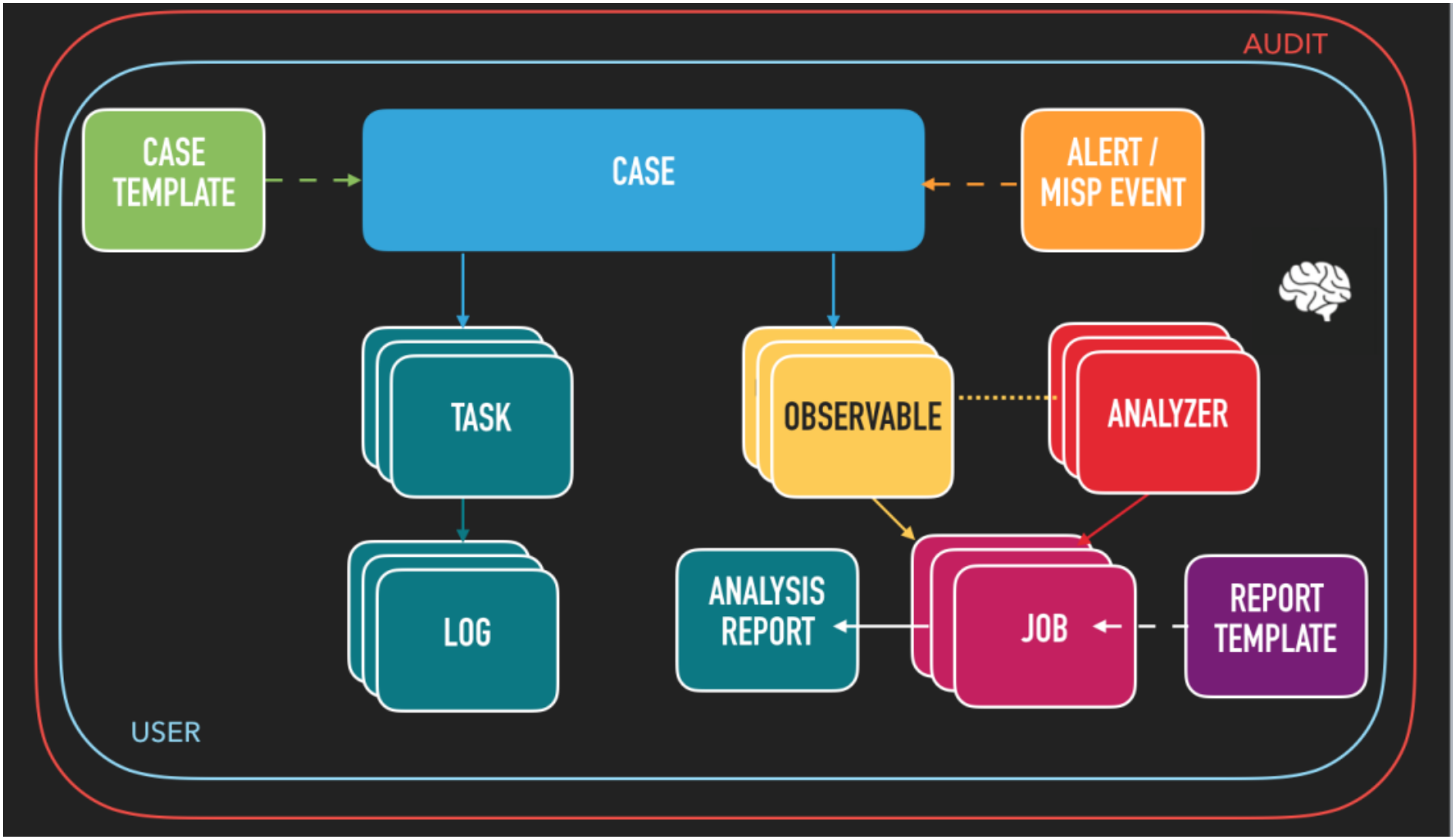


Table of contents

Features of TheHive

1. **Case Management:**
   - Create cases to document and manage incidents.
   - Assign cases to analysts or teams for resolution.
2. **Task Assignment:**
   - Assign tasks and responsibilities within cases.
   - Ensure organized and efficient incident response.
3. **Alerts and Notifications:**
   - Receive alerts to notify analysts of potential security incidents.
   - Stay informed and respond promptly.

Features of TheHive

1. **Case Management:**
   - Create cases to document and manage incidents.
   - Assign cases to analysts or teams for resolution.
2. **Task Assignment:**
   - Assign tasks and responsibilities within cases.
   - Ensure organized and efficient incident response.
3. **Alerts and Notifications:**
   - Receive alerts to notify analysts of potential security incidents.
   - Stay informed and respond promptly.

Features of TheHive

1. **Case Management:**
   - Create cases to document and manage incidents.
   - Assign cases to analysts or teams for resolution.
2. **Task Assignment:**
   - Assign tasks and responsibilities within cases.
   - Ensure organized and efficient incident response.
3. **Alerts and Notifications:**
   - Receive alerts to notify analysts of potential security incidents.
   - Stay informed and respond promptly.

Features of TheHive (Contd)

4. **Observable Analysis:**
   - Document and analyze potential threats with observables.
   - Utilize Cortex for real-time observable analysis.
5. **Analyzer Integration:**
   - Use various analyzers to gather additional information.
   - Enhance incident response with automated analysis.
6. **Information Sharing:**
   - Tight integration with MISP for sharing threat intelligence.
   - Import/export cases and synchronize observables with MISP.

Features of TheHive (Contd)

4. **Observable Analysis:**
   - Document and analyze potential threats with observables.
   - Utilize Cortex for real-time observable analysis.
5. **Analyzer Integration:**
   - Use various analyzers to gather additional information.
   - Enhance incident response with automated analysis.
6. **Information Sharing:**
   - Tight integration with MISP for sharing threat intelligence.
   - Import/export cases and synchronize observables with MISP.

Features of TheHive (Contd)

4. **Observable Analysis:**
   - Document and analyze potential threats with observables.
   - Utilize Cortex for real-time observable analysis.
5. **Analyzer Integration:**
   - Use various analyzers to gather additional information.
   - Enhance incident response with automated analysis.
6. **Information Sharing:**
   - Tight integration with MISP for sharing threat intelligence.
   - Import/export cases and synchronize observables with MISP.

Table of contents

Two key Features

- Case management
- Observable analysis with Cortex integration

Two key Features

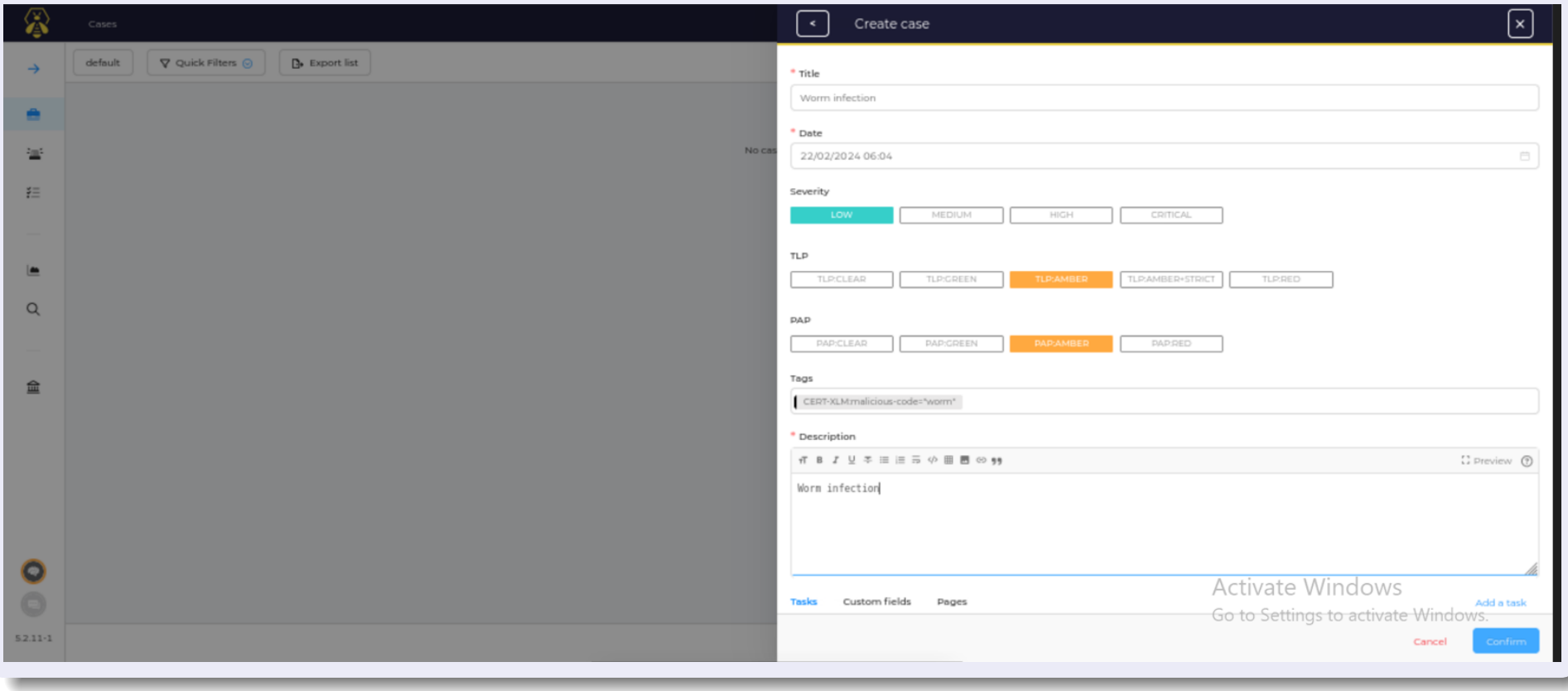
- Case management
- Observable analysis with Cortex integration

Demonstration

Live Demonstration

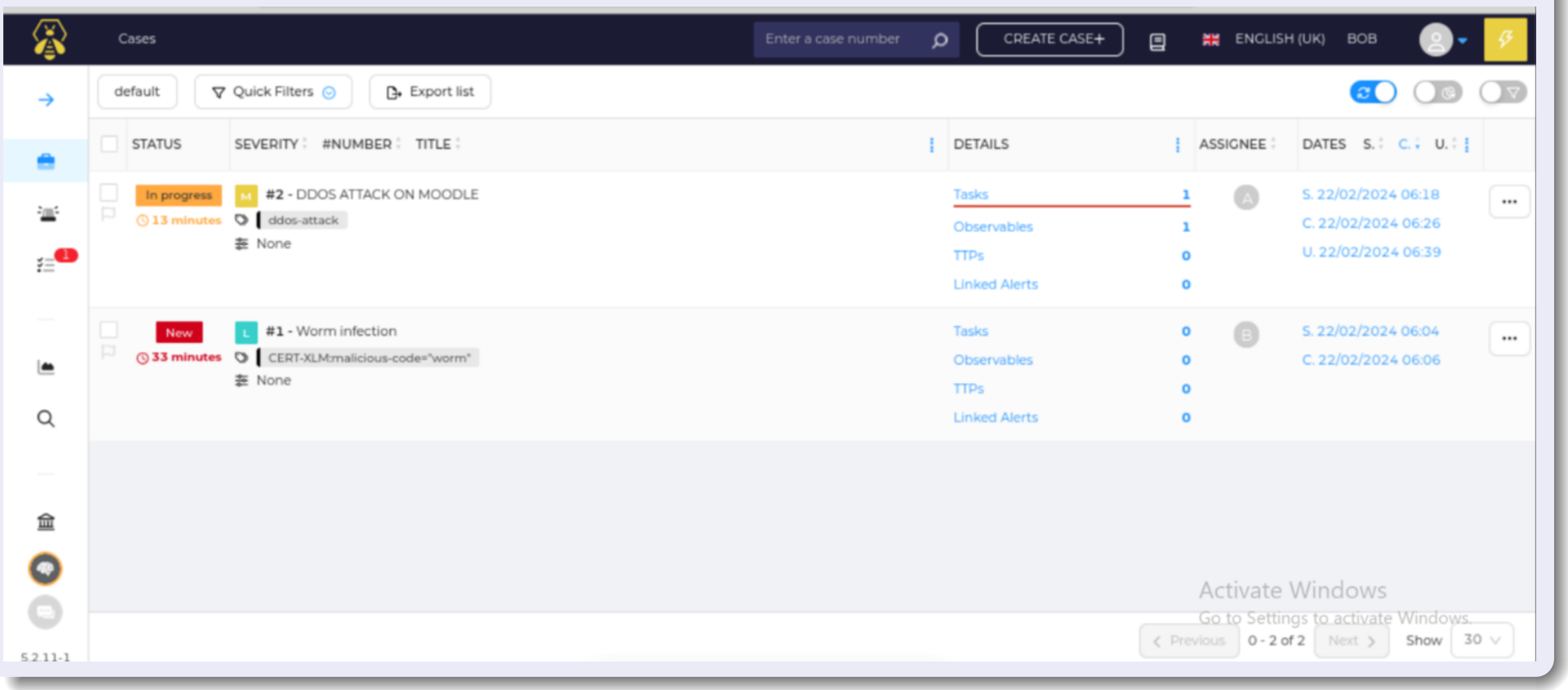
Demonstration of feature One

Creating A case



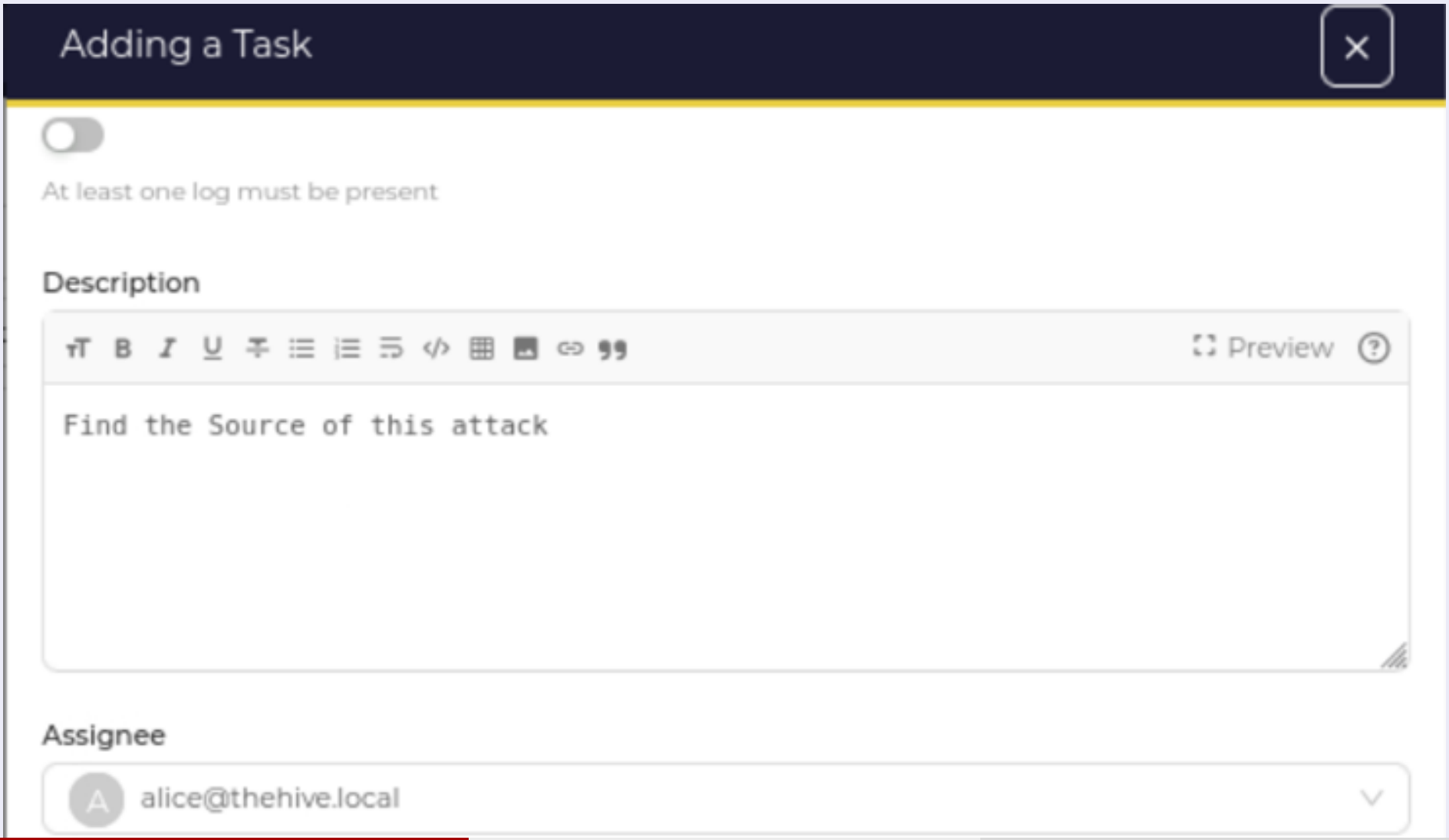
Demonstration of feature One (Contd.)

After Successful Creation of A case



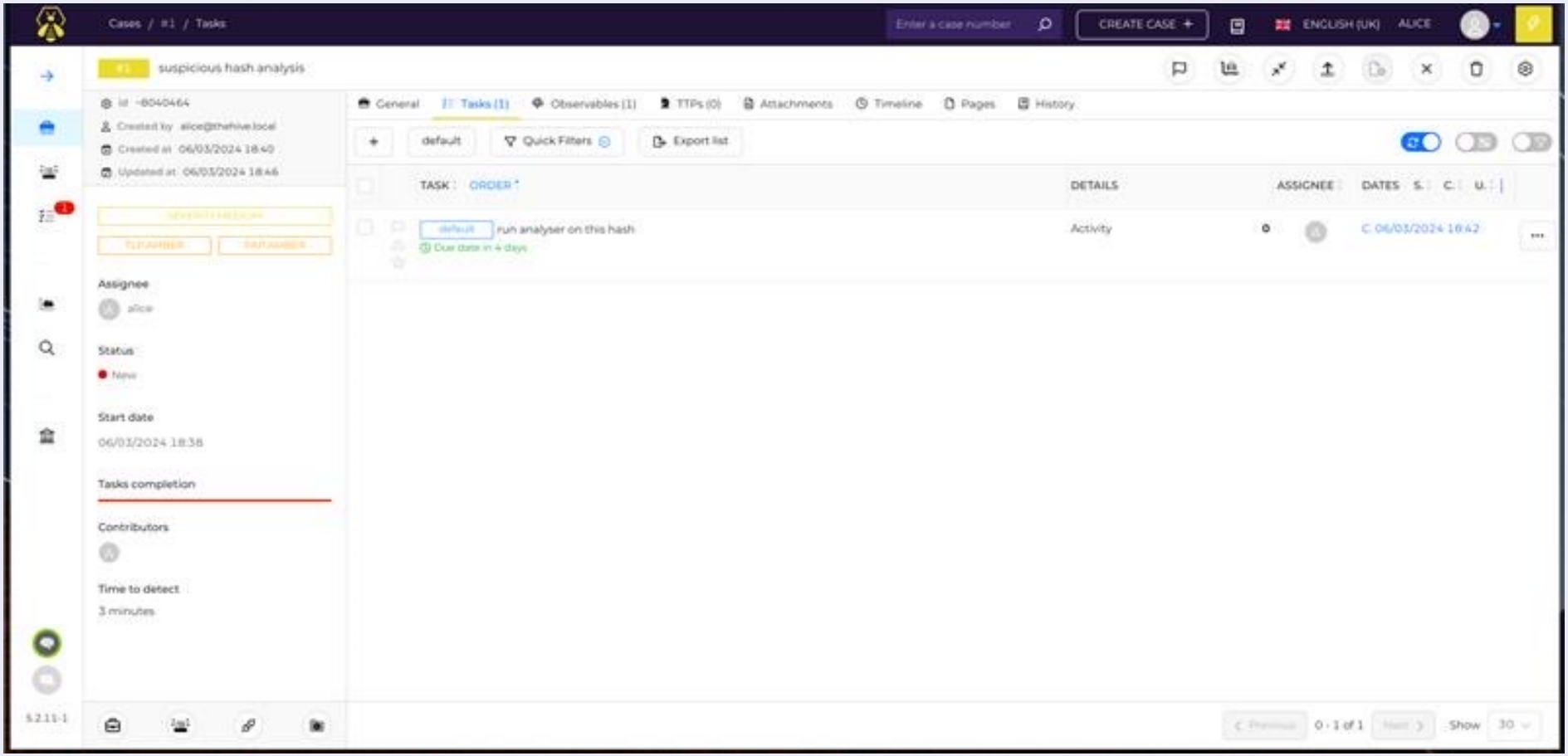
Demonstration of feature One (Contd.)

Adding a Task to a case and assign a User to it



Demonstration of feature One (Contd.)

After Task Assignment



Demonstration of Feature Two :Observables

- Observables are pieces of information related to a security incident.
- Observables can be added to cases in TheHive.
- Observables have different types, such as url, mail subject, or registry key.
- Here the user (security analysts) will report their analysis.(e.g. IP address , hash of malicious files)

Demonstration of Feature Two :Observables

- Observables are pieces of information related to a security incident.
- Observables can be added to cases in TheHive.
- Observables have different types, such as url, mail subject, or registry key.
- Here the user (security analysts) will report their analysis.(e.g. IP address , hash of malicious files)

Demonstration of Feature Two :Observables

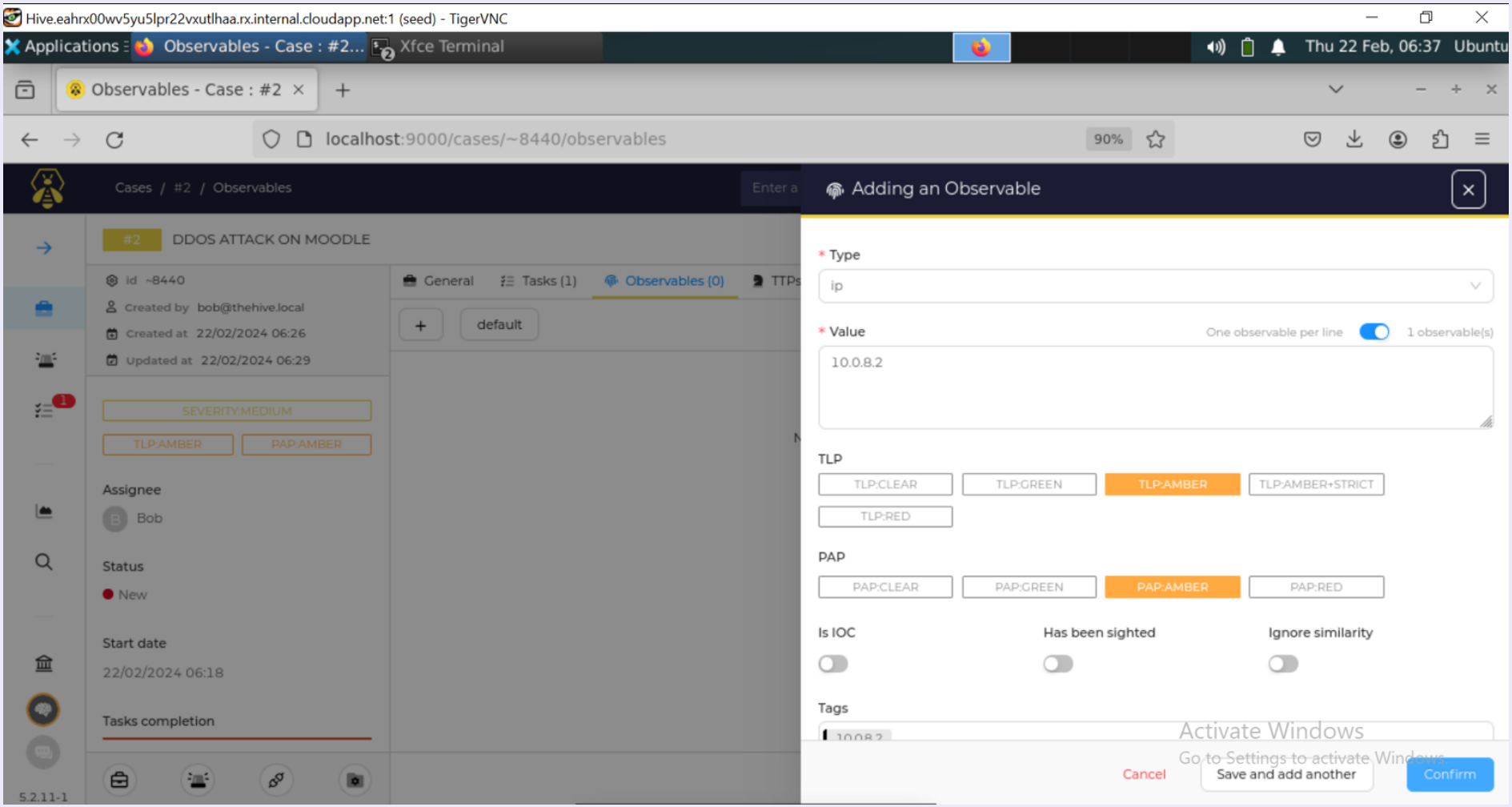
- Observables are pieces of information related to a security incident.
- Observables can be added to cases in TheHive.
- Observables have different types, such as url, mail subject, or registry key.
- Here the user (security analysts) will report their analysis.(e.g. IP address , hash of malicious files)

Demonstration of Feature Two :Observables

- Observables are pieces of information related to a security incident.
- Observables can be added to cases in TheHive.
- Observables have different types, such as url, mail subject, or registry key.
- Here the user (security analysts) will report their analysis.(e.g. IP address , hash of malicious files)

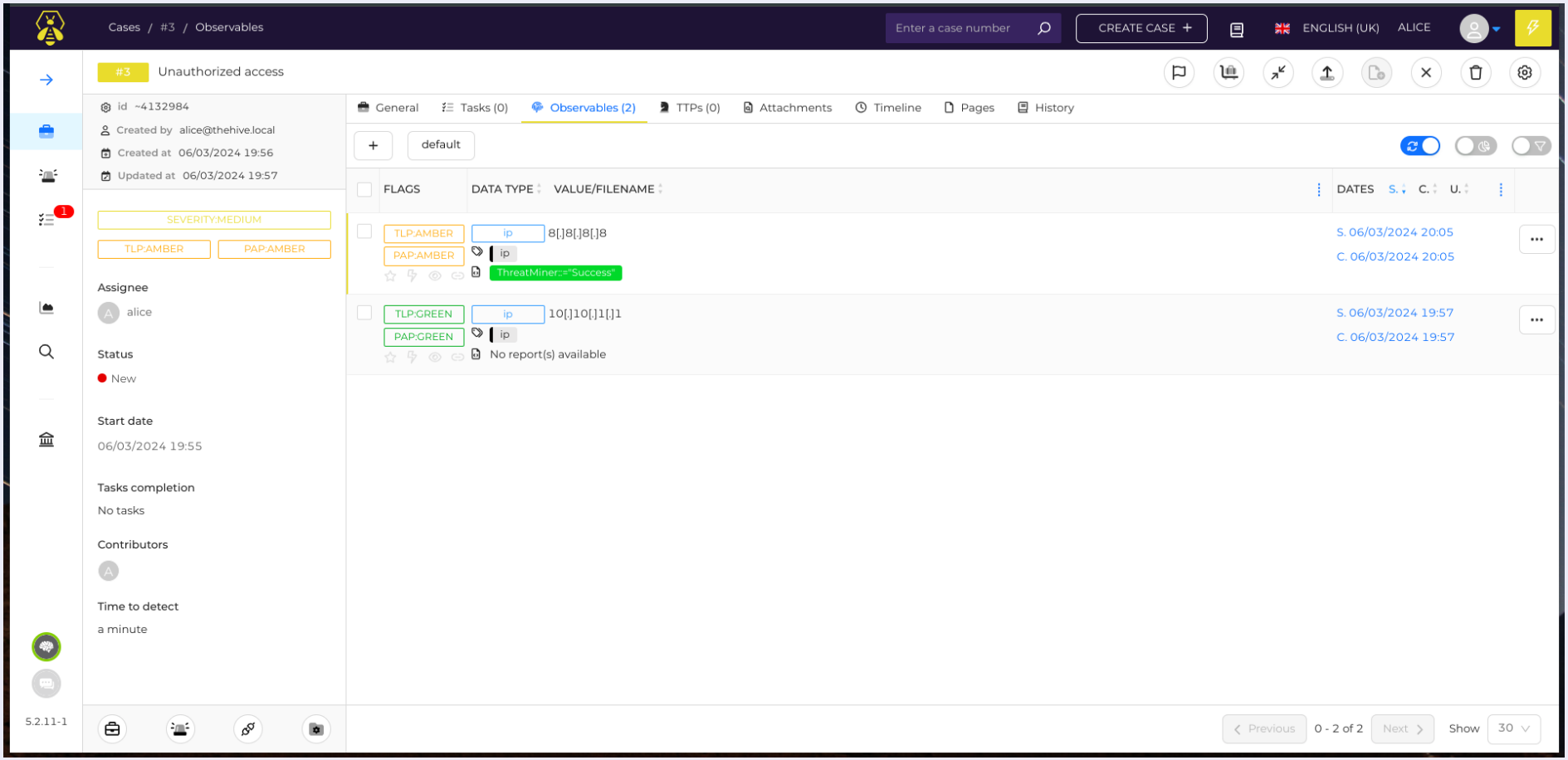
Demonstration of feature Two (Contd.)

Creating An Observabale



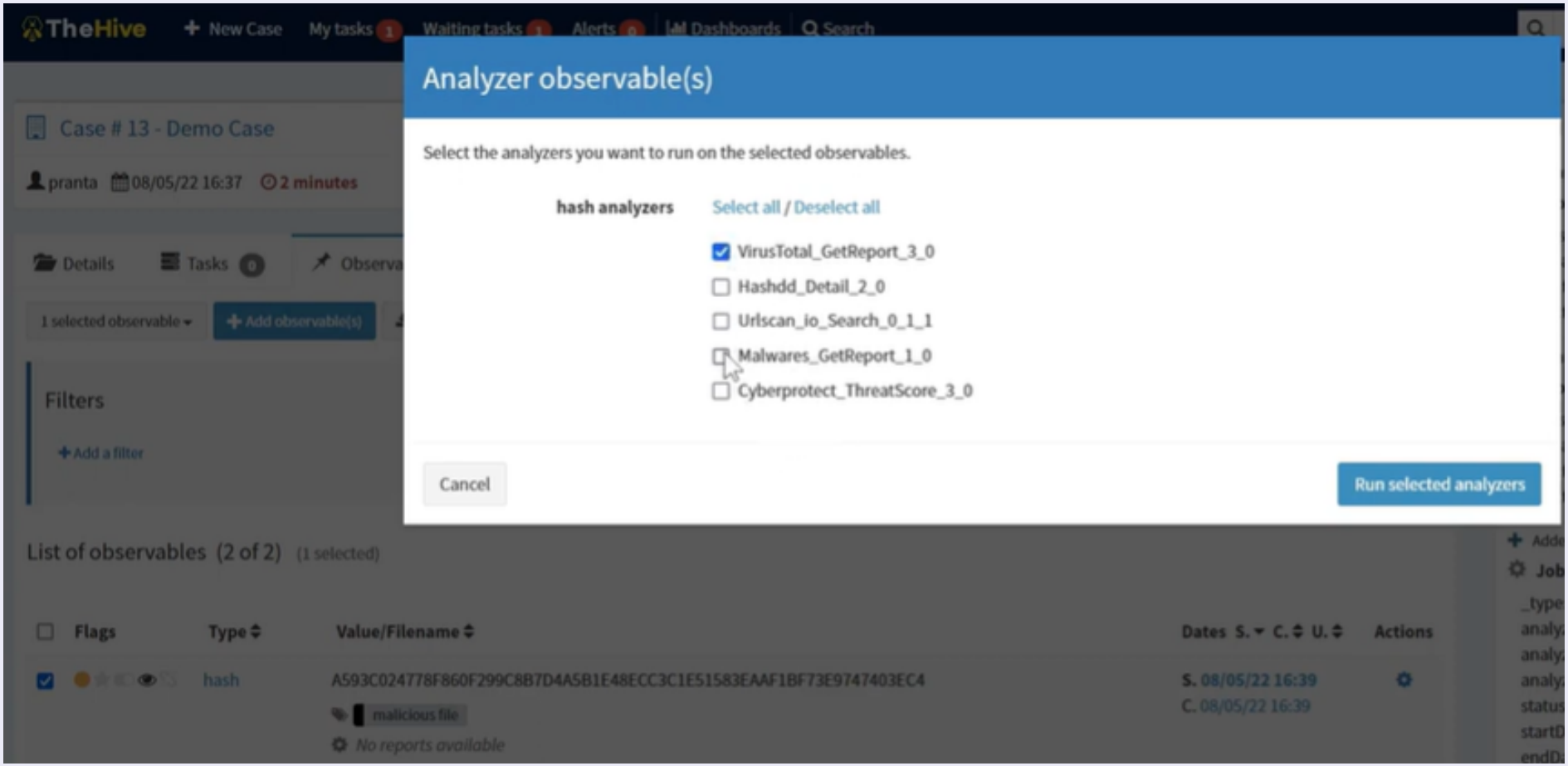
Demonstration of feature Two (Contd.)

Finally we can See the observables in a Case Like this



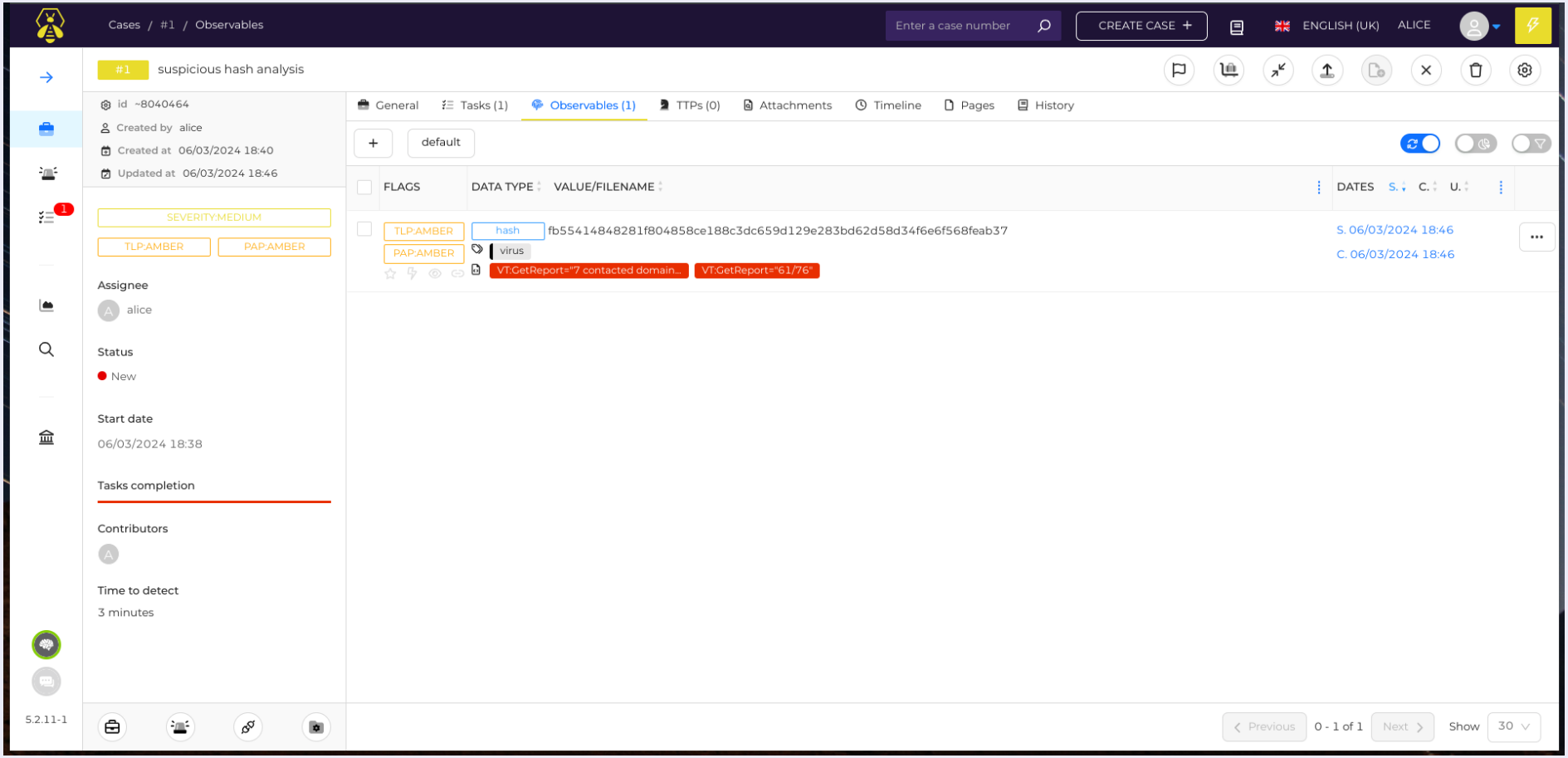
Demonstration of feature Two (Contd.)

Applying analyzer



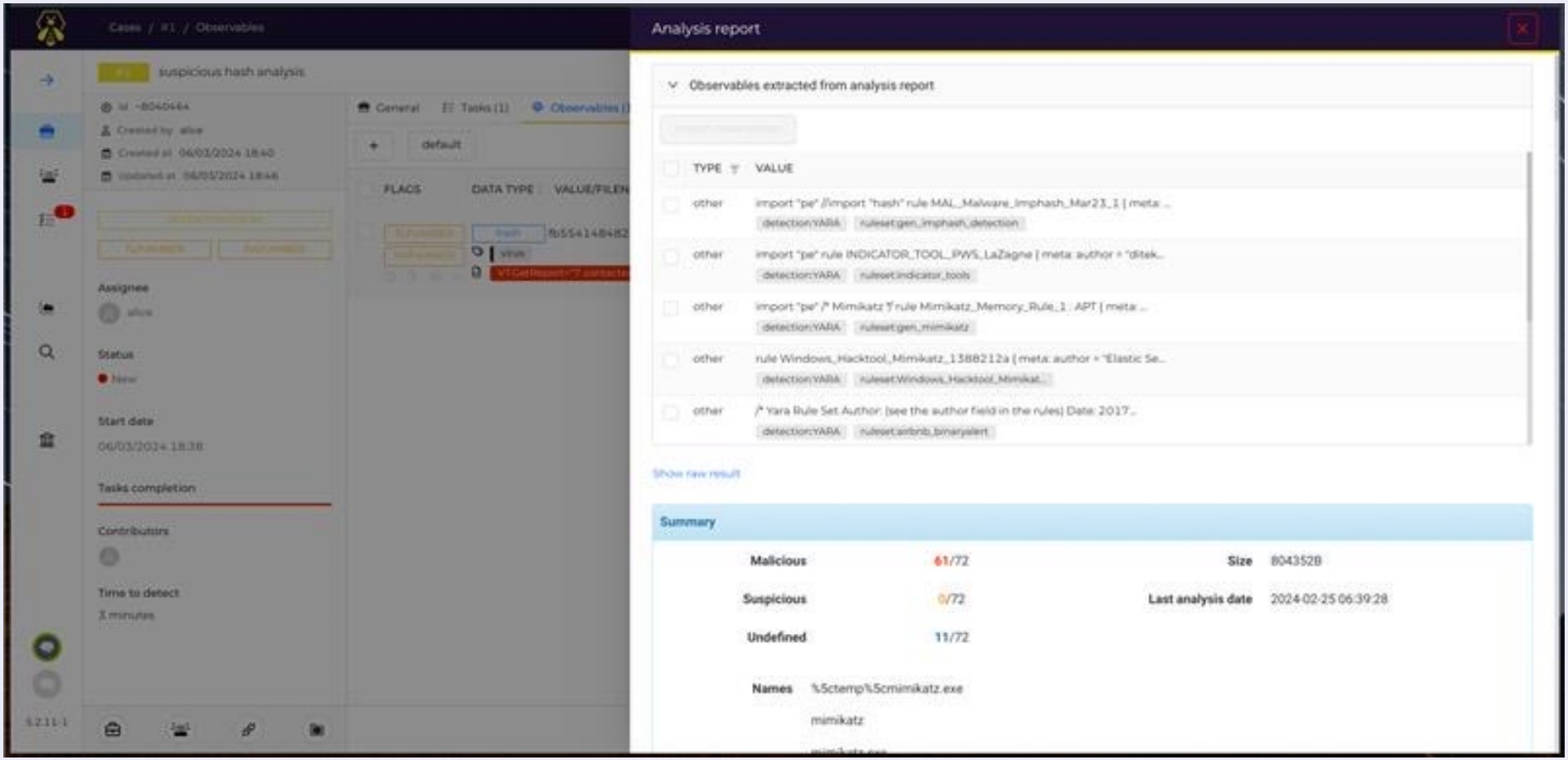
Demonstration of feature Two (Contd.)

Analysis result



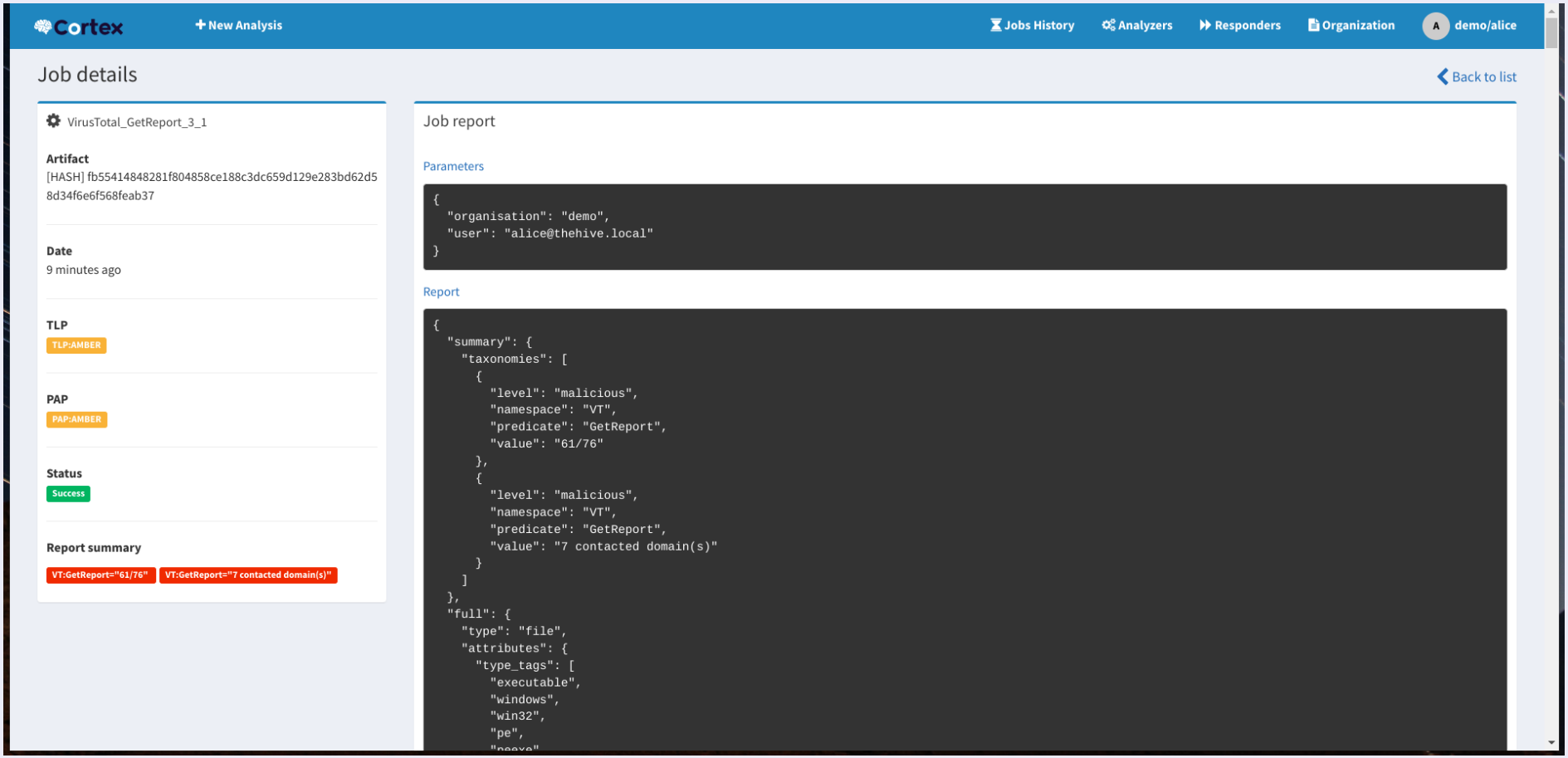
Demonstration of feature Two (Contd.)

Analysis report



Demonstration of feature Two (Contd.)

Job details in Cortex(Job report in JSON format)



Summary

- TheHive is a collaborative Security Incident Response Platform.
- Features include case management, observable analysis, and active response.
- The architecture comprises a sleek frontend, robust backend, Cortex analytics, reliable storage, and powerful analyzers.
- Key concepts involve users, organizations, cases, and tasks.
- Analyzers enhance intelligence, while responders enable active responses.

Thank You!

Thank you for your attention!

Any questions or discussions are welcome.