Strong User Authentication using FIDO and Java Card

Vlad Petrovici
Software Engineer
Oracle - Java Platform Group

Nicolae Bors
Software Engineer
Oracle - Java Platform Group

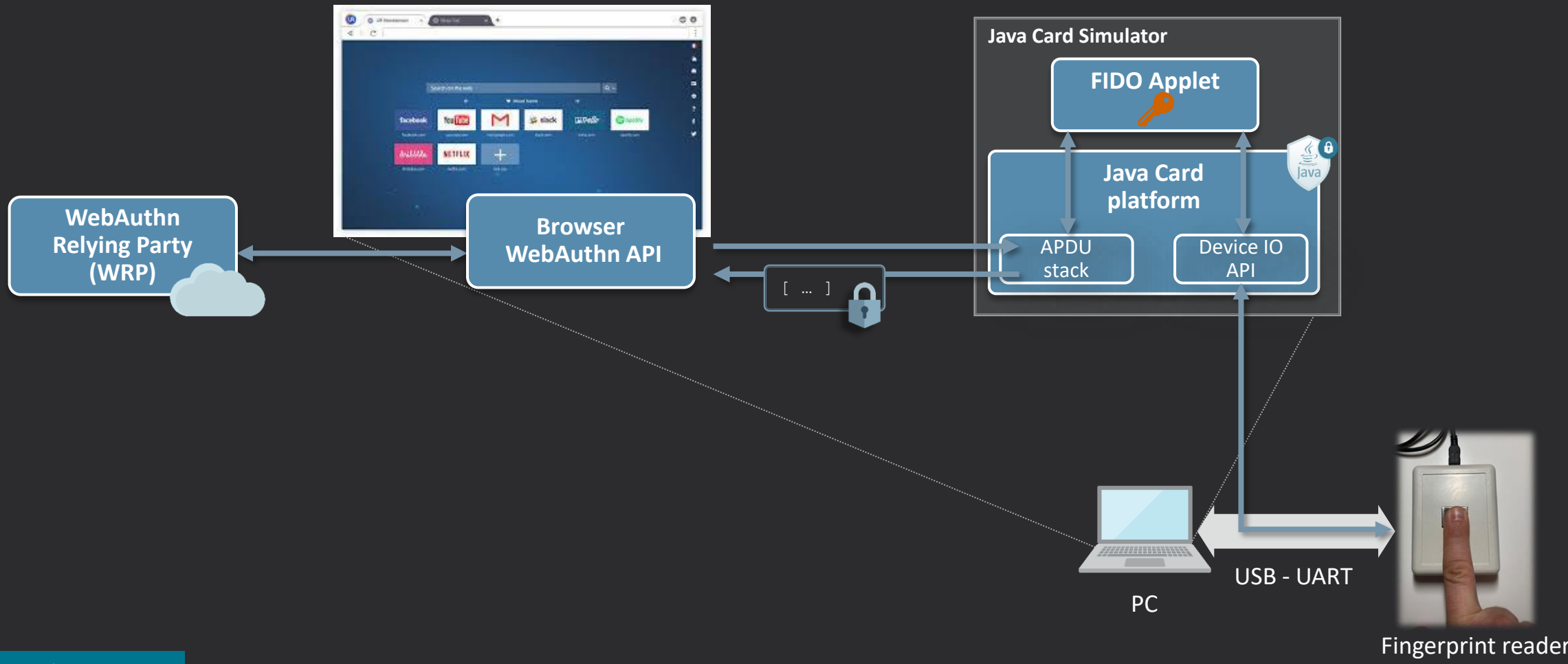
Oracle Groundbreakers

ORACLE

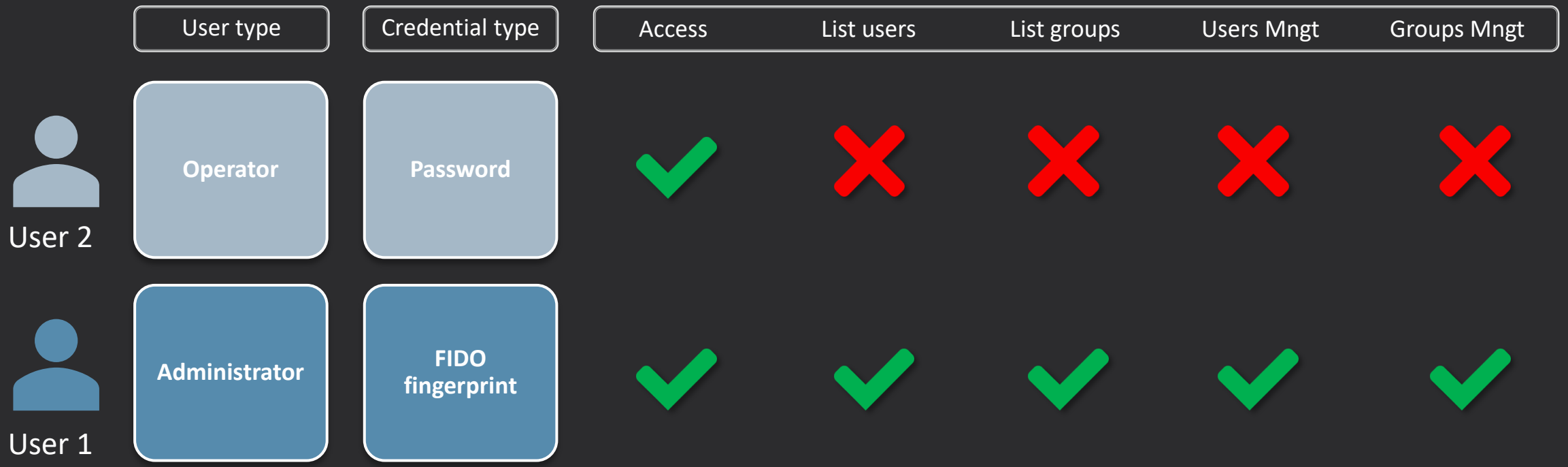
Safe Harbor Statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

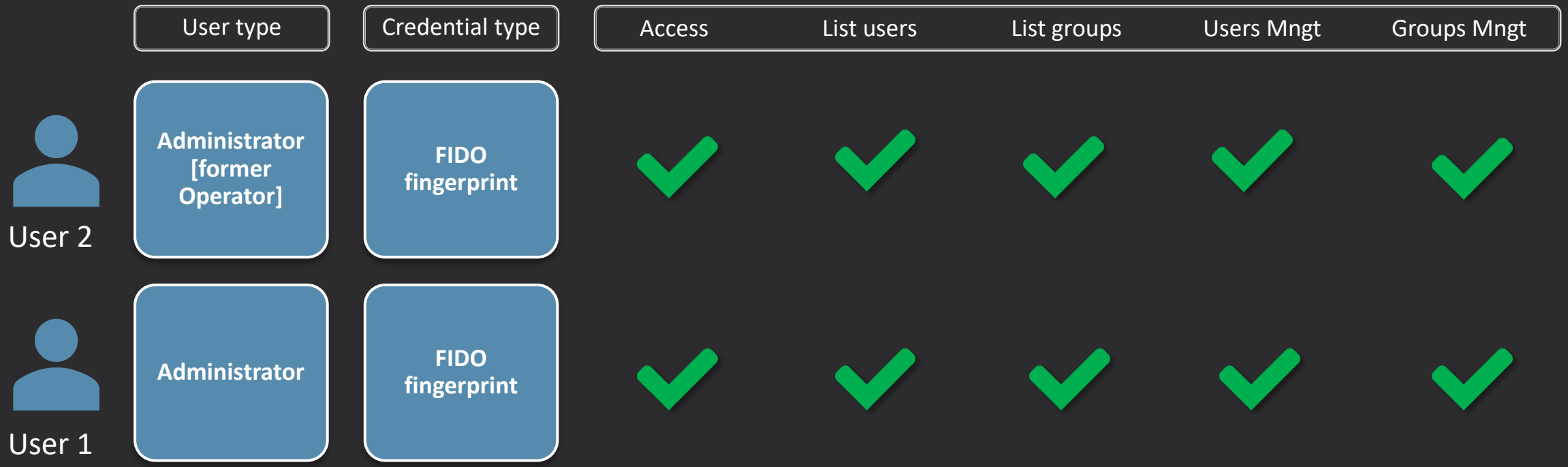
Demo
Architecture



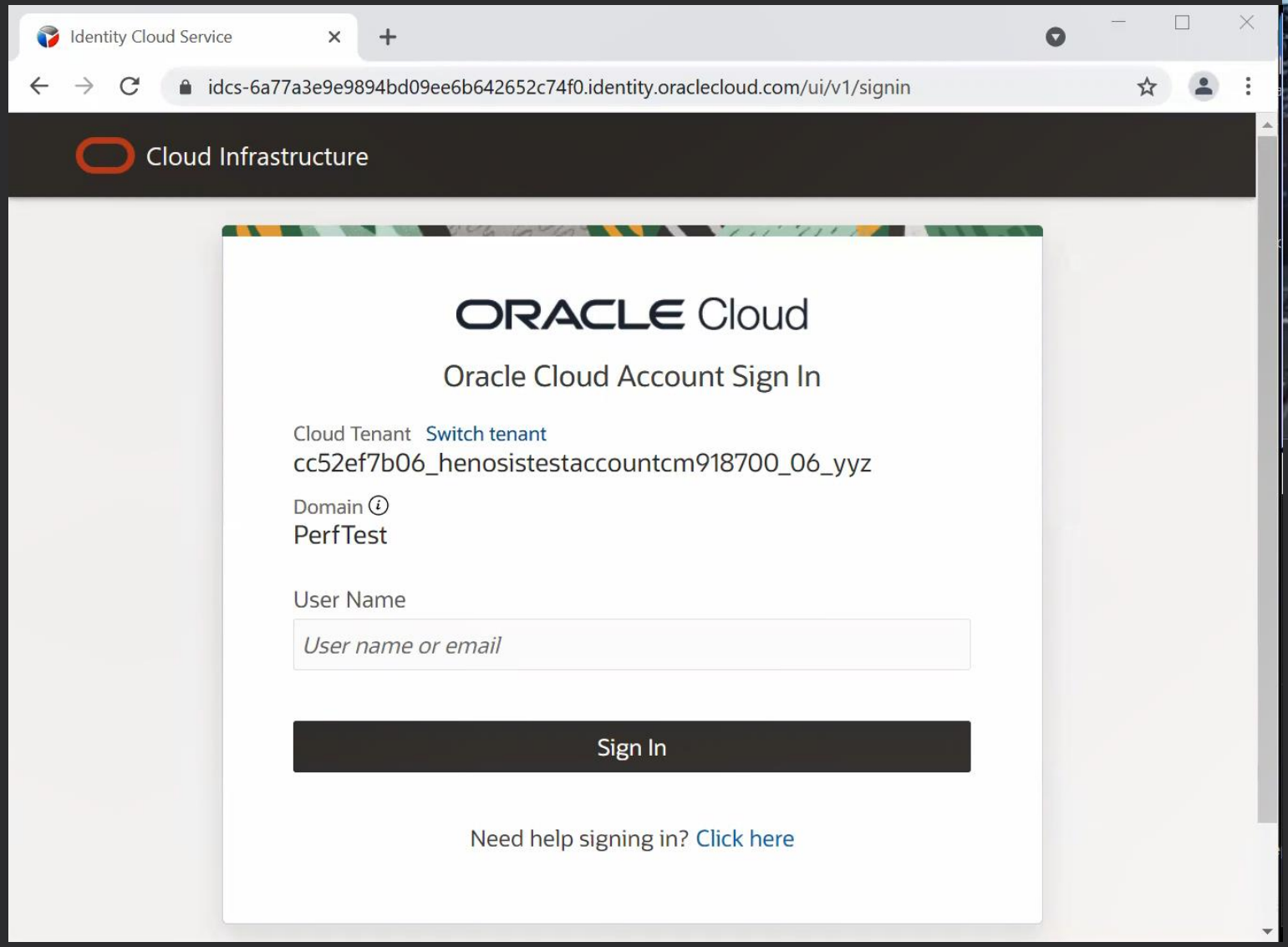
Demo
Use Case

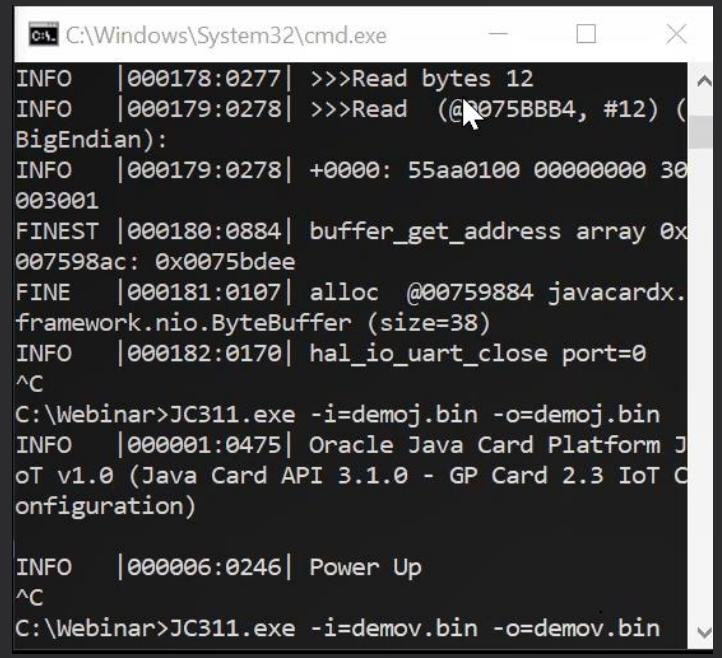


Demo
Use Case



Demo





Demo

FIDO using Java Card

Conclusion

Secure Runtime

- To securely store and manage crypto keys for FIDO
- To run the FIDO service in the Secure Element: retrieve public key, build responses and sign them.

Portable

- To address the highly fragmented IoT landscape
- To deploy and operate the service on multiple hardware platforms, from different vendors, at lower cost

Adaptable & Extensible

- To support multiple implementation applications e.g. FIDO, IoT-SAFE, etc.
- To enable payload handling from different peripherals using various protocols
- Able to use Biometry using Java Card API

Manageable

- To update and upgrade the Java Card applets and remaining compliant with the fast-evolving security requirements and regulations

FIDO using Java Card

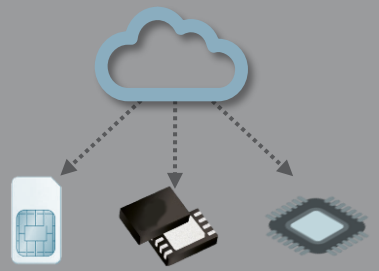
Conclusion

Secure Runtime

- To securely store and manage crypto keys for FIDO
- To run the FIDO service in the Secure Element: retrieve public key, build responses and sign them.

Portable

- To address the highly fragmented IoT landscape
- To deploy and operate the service on multiple hardware platforms, from different vendors, at lower cost



Adaptable & Extensible

- To support multiple implementation applications e.g. FIDO, IoT-SAFE, etc.
- To enable payload handling from different peripherals using various protocols
- Able to use Biometry using Java Card API

Manageable

- To update and upgrade the Java Card applets and remaining compliant with the fast-evolving security requirements and regulations

FIDO using Java Card

Conclusion

Secure Runtime

- To securely store and manage crypto keys for FIDO
- To run the FIDO service in the Secure Element: retrieve public key, build responses and sign them.

Portable

- To address the highly fragmented IoT landscape
- To deploy and operate the service on multiple hardware platforms, from different vendors, at lower cost

Adaptable & Extensible

- To support multiple implementation applications e.g. FIDO, IoT-SAFE, etc.
- To enable payload handling from different peripherals using various protocols
- Able to use Biometry using Java Card API

Manageable

- To update and upgrade the Java Card applets and remaining compliant with the fast-evolving security requirements and regulations

FIDO using Java Card

Conclusion

Secure Runtime

- To securely store and manage crypto keys for FIDO
- To run the FIDO service in the Secure Element: retrieve public key, build responses and sign them.

Portable

- To address the highly fragmented IoT landscape
- To deploy and operate the service on multiple hardware platforms, from different vendors, at lower cost

Adaptable & Extensible

- To support multiple implementation applications e.g. FIDO, IoT-SAFE, etc.
- To enable payload handling from different peripherals using various protocols
- Able to use Biometry using Java Card API

Manageable

- To update and upgrade the Java Card applets and remaining compliant with the fast-evolving security requirements and regulations

FIDO using Java Card

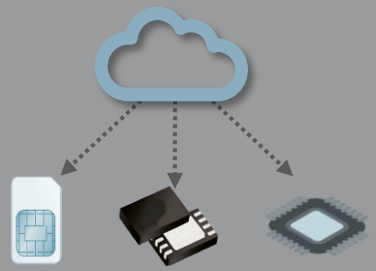
Conclusion

Secure Runtime

- To securely store and manage crypto keys for FIDO
- To run the FIDO service in the Secure Element: retrieve public key, build responses and sign them.

Portable

- To address the highly fragmented IoT landscape
- To deploy and operate the service on multiple hardware platforms, from different vendors, at lower cost

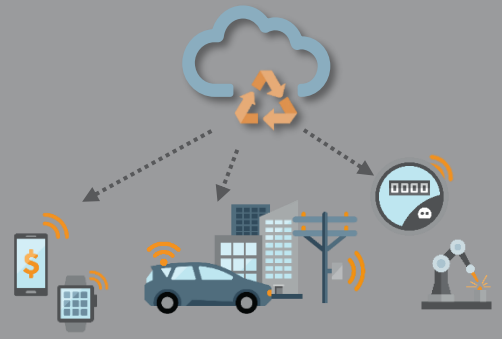


Adaptable & Extensible

- To support multiple implementation applications e.g. FIDO, IoT-SAFE, etc.
- To enable payload handling from different peripherals using various protocols
- Able to use Biometry using Java Card API

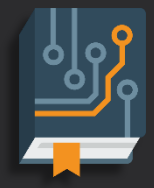
Manageable

- To update and upgrade the Java Card applets and remaining compliant with the fast-evolving security requirements and regulations



More Information

https://www.oracle.com/java/technologies/java-card-tech.html



Java Card Platform Specification 3.1

Latest release of the Java Card specification and the reference for Java Card products.

Java Card Development Kit Tools

The Java Card Development Kit Tools are used to convert and verify Java Card applications. The Tools can be used with products based on version 3.1, 3.0.5 and 3.0.4 of the Java Card Specifications.

Java Card Development Kit Simulator

The Java Card Development Kit Simulator includes a simulation component and Eclipse plug-in.
Combined with the Java Card Development Kit Tools, it provides a complete, stand-alone development environment.

Java Card IoT and Security blog

This Blog covers the latest Java technology for small devices and security in the IoT, mobile, ID and Payment.

Webcast – Secure Business Runs Java Card

Webcast – How to secure IoT Edge with Java Card

Webcast: Oracle Java Card 3.1 Boosts Security for IoT Devices at the Edge

contacts: nicolae.bors at oracle.com, vlad.petrovici at oracle.com

Questions ?