Virtualization + Cloud & Security

Mgr. Michael Grafnetter

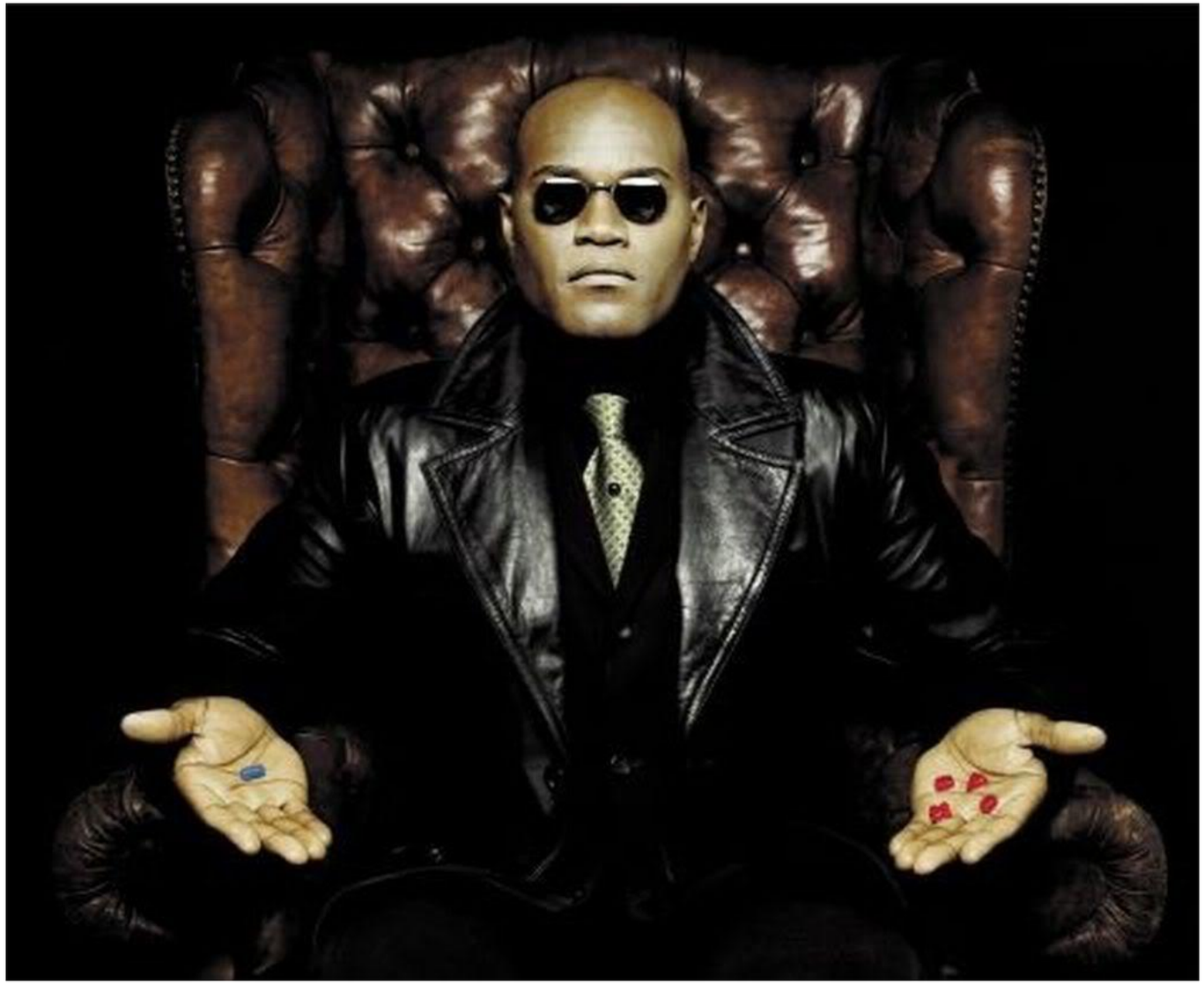
Agenda

- Virtualization Security Risks and Solutions
- Cloud Computing Security Risks
- Authentication in Cloud Applications

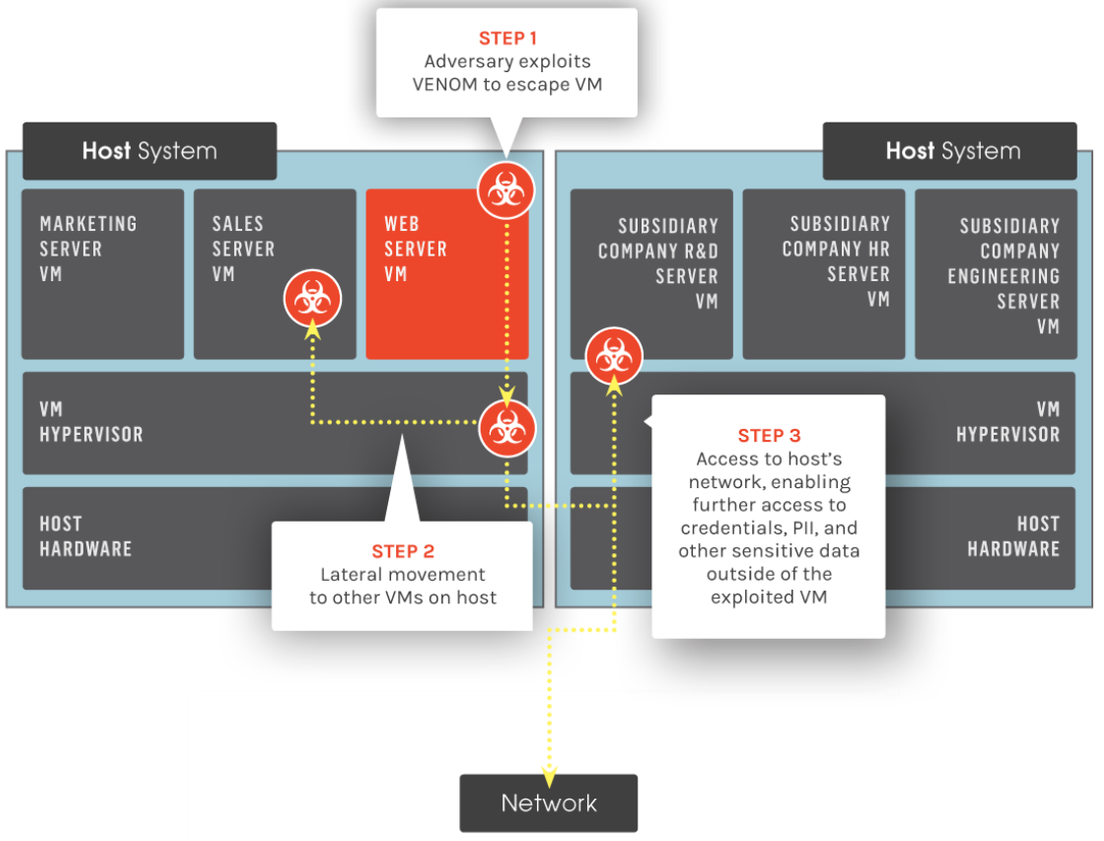
dsinternals.com/nswi150

Virtualization Security Risks

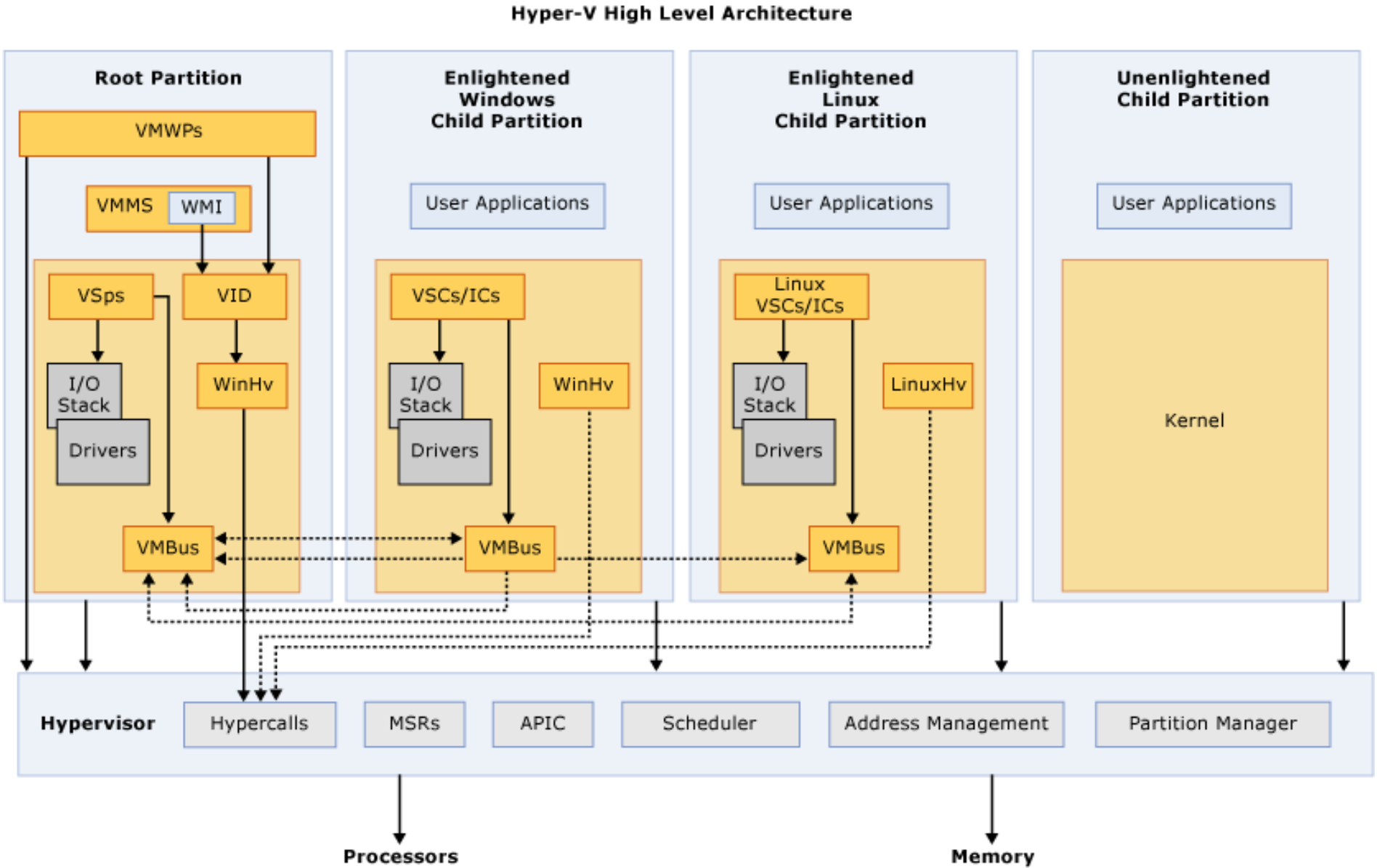
Blue Pill Attack



VM Escape Vulnerabilities



VM Escape Vulnerabilities

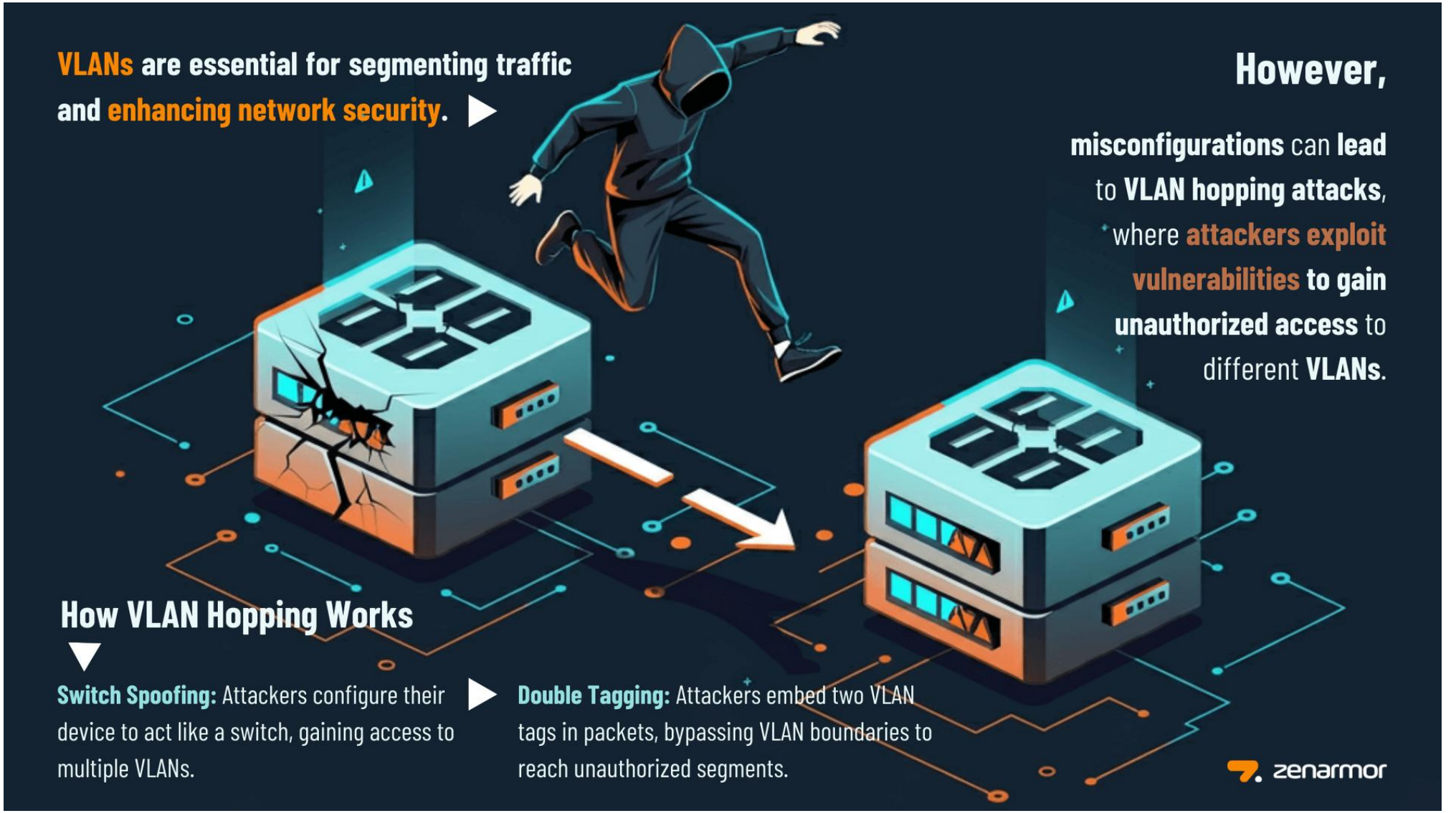


Known VM Escape Vulnerabilities

CVE-2007-1744	VMware Workstation
CVE-2008-0923	VMware Workstation
CVE-2009-1244	VMware ESXi, Workstation, Fusion
CVE-2012-0217	Xen Hypervisor
CVE-2014-0983	Oracle VirtualBox
CVE-2015-3456	QEMU
CVE-2015-7835	Xen Hypervisor
CVE-2016-6258	Xen Hypervisor
CVE-2016-7092	Xen Hypervisor
CVE-2017-0075	Microsoft Hyper-V
CVE-2017-0109	Microsoft Hyper-V
CVE-2017-4903	VMware ESXi, Workstation, Fusion
CVE-2017-4934	VMware Workstation, Fusion
CVE-2017-4936	VMware Workstation, Horizon View
CVE-2018-2698	Oracle VirtualBox

CPU Vulnerabilities

VLAN Hopping

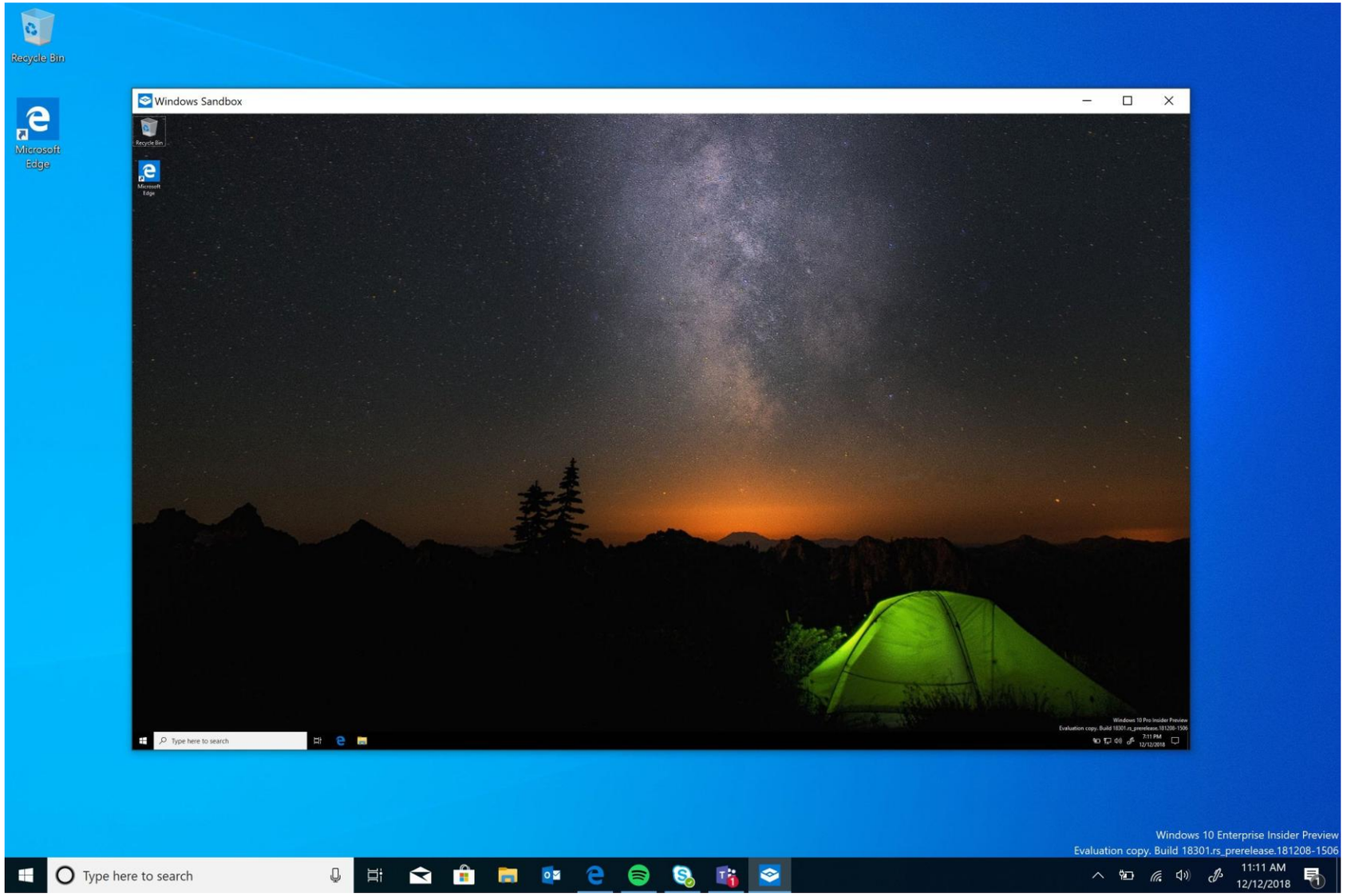


Other Risks of Virtualization

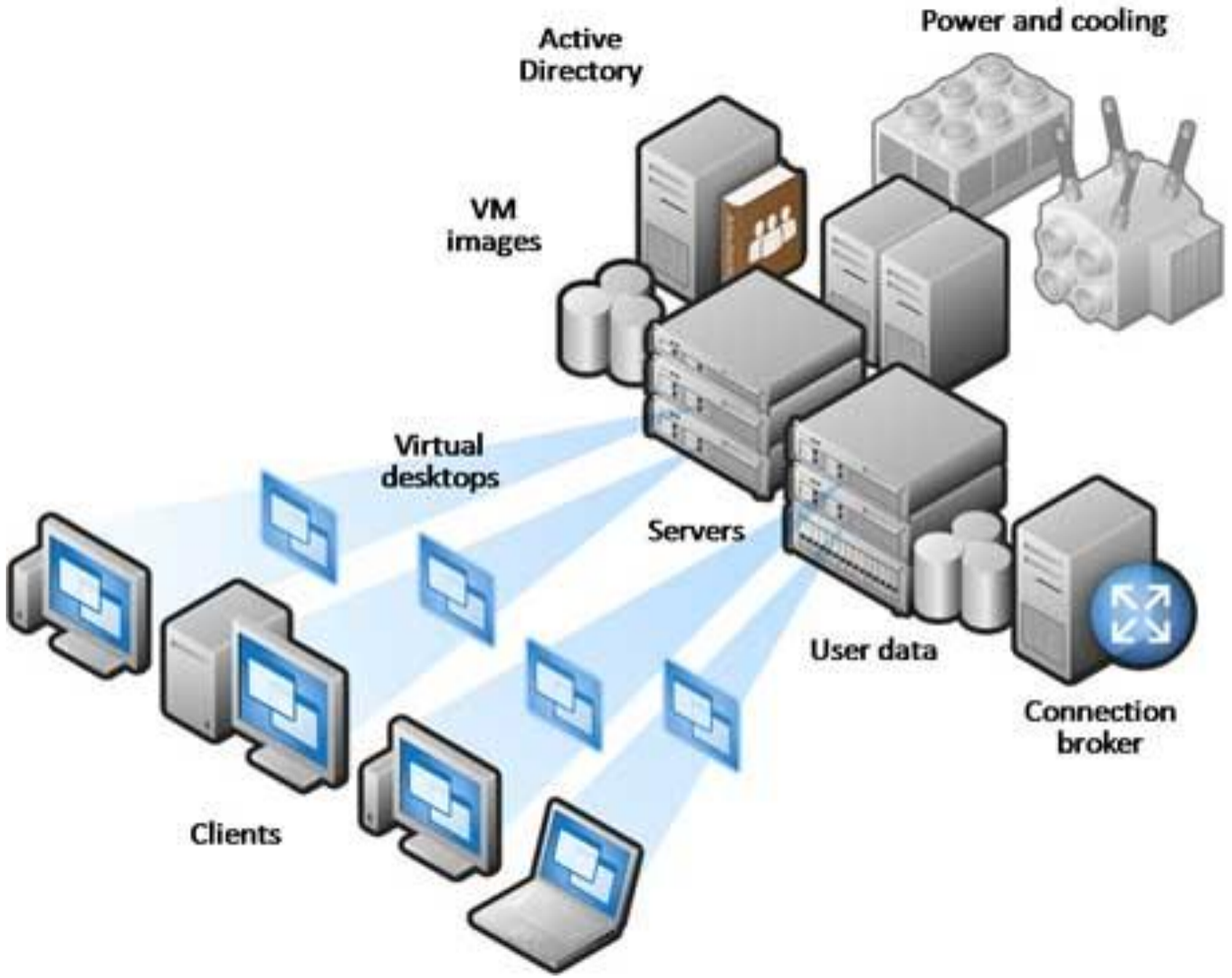
- Introduction of yet another OS
- Reliance on traditional barriers
- Accelerated provisioning
- Security left to non-traditional security staff
- Audit scope creep

Virtualization-Based Security Solutions

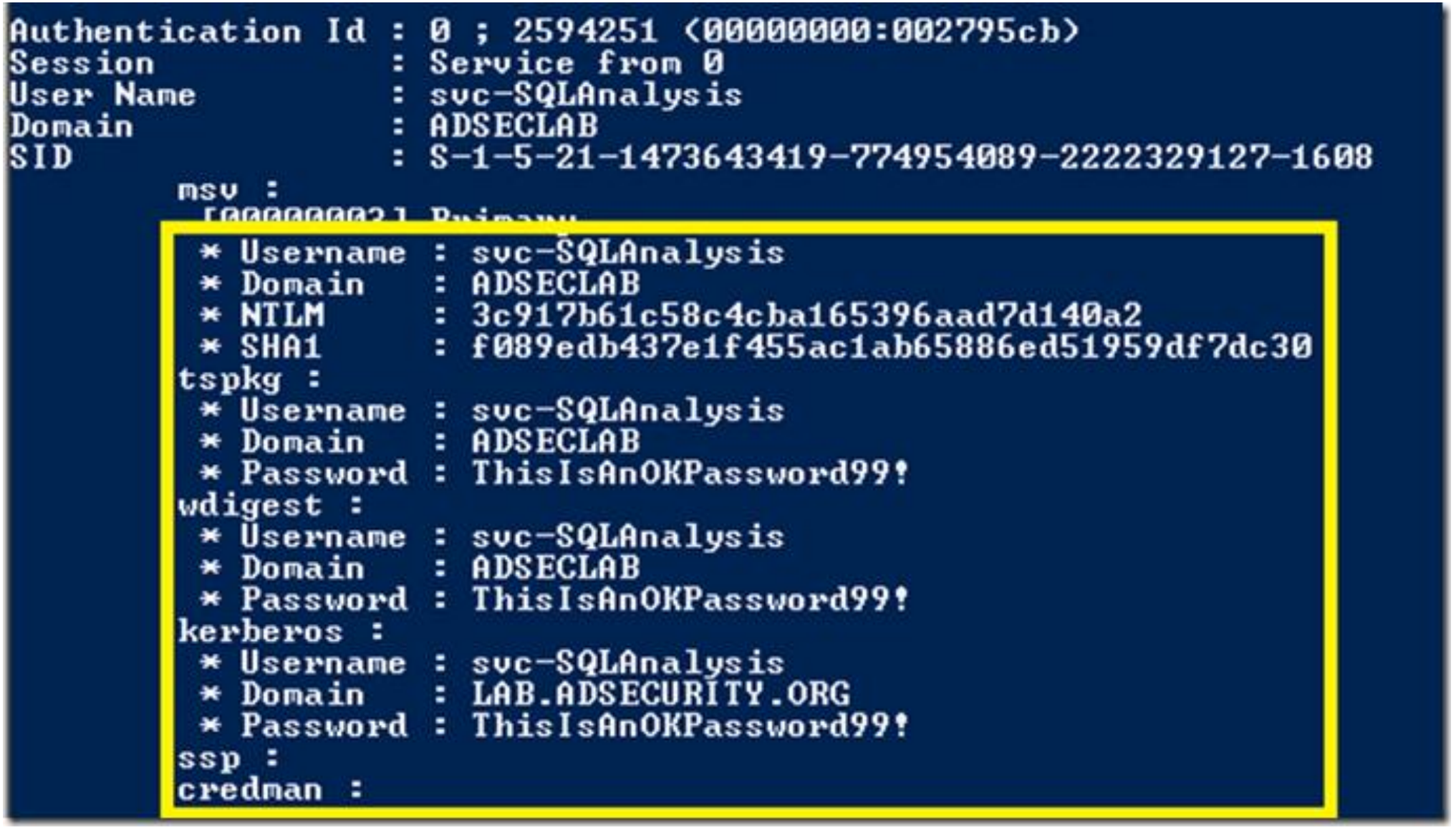
Sandboxing


Recycle Bin
Microsoft Edge
Windows Sandbox
Recycle Bin
Microsoft Edge
Type here to search
Windows 10 Pro Insider Preview
Evaluation copy. Build 18301.rs_prerelease.181208-1506
7:11 PM
12/12/2018
Type here to search
Windows 10 Enterprise Insider Preview
Evaluation copy. Build 18301.rs_prerelease.181208-1506
11:11 AM
12/12/2018

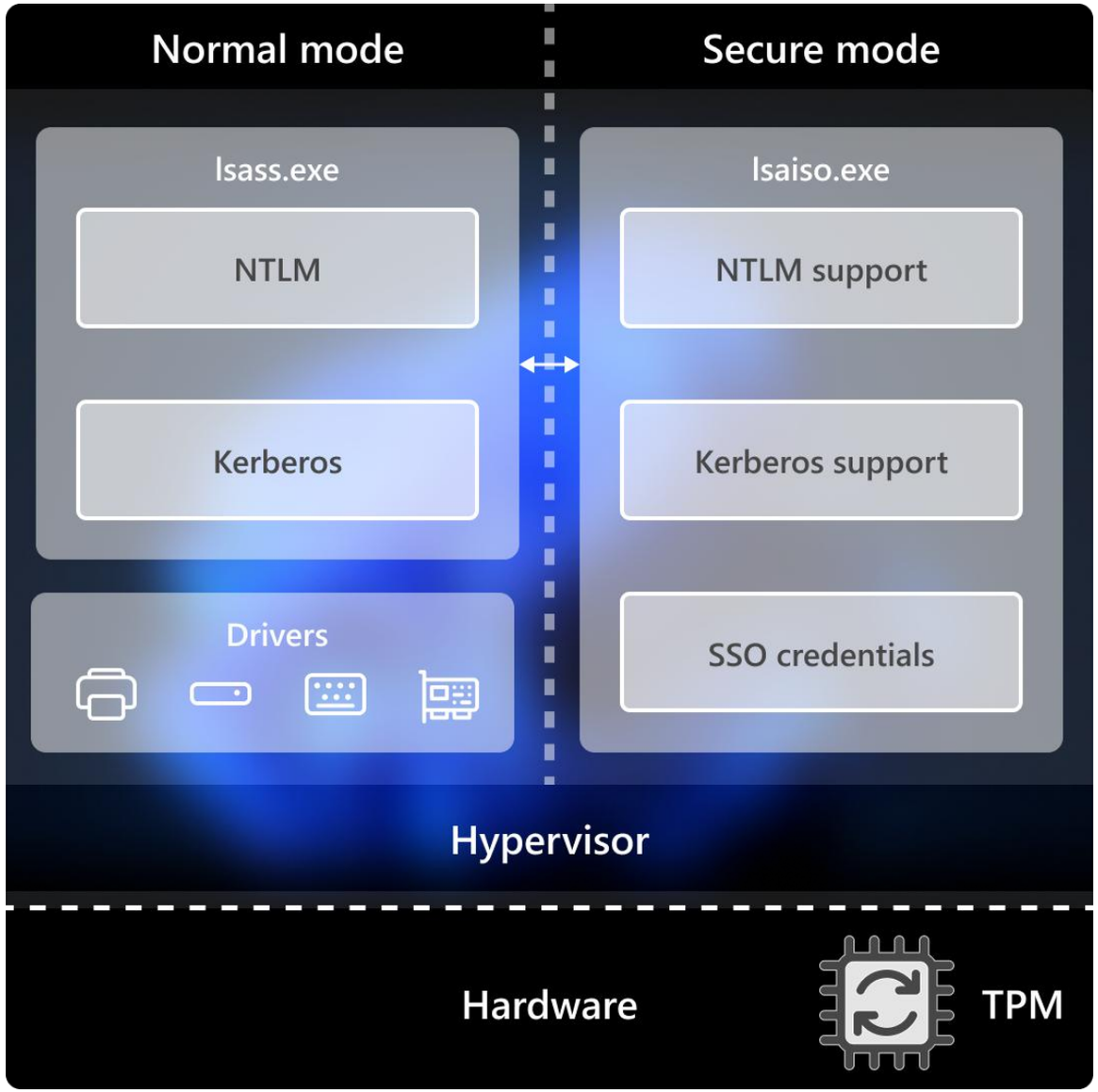
Virtual Desktop Infrastructure



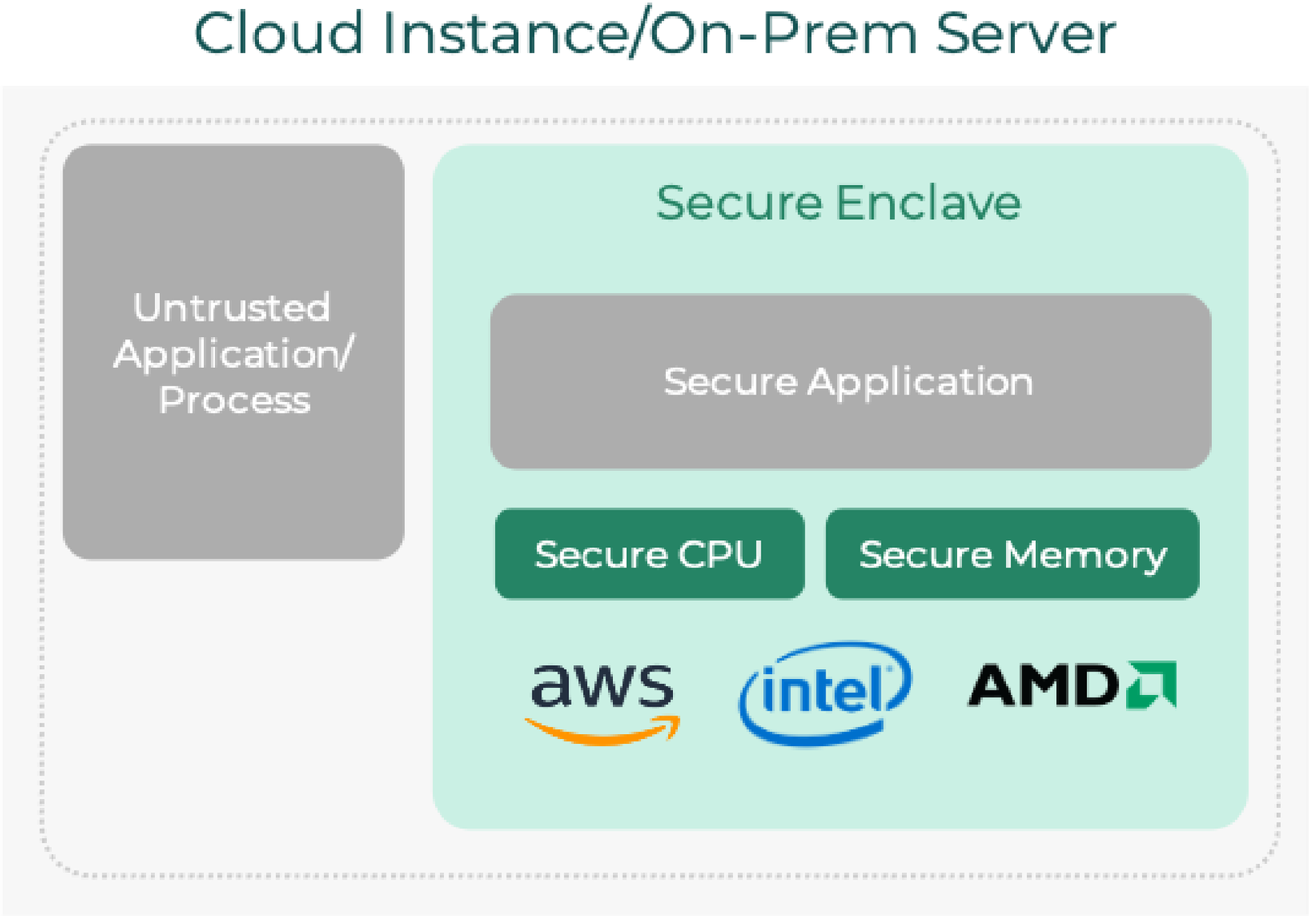
Windows 10+ Virtual Secure Mode: Motivation



Windows 10+ Virtual Secure Mode



Secure Enclaves / Trusted Computing



Cloud Computing Security Risks

Who has access to our data?

Example: Azure Datacenter Security

24 hour monitored
PHYSICAL SECURITY

Centralized
MONITORING AND ALERTS

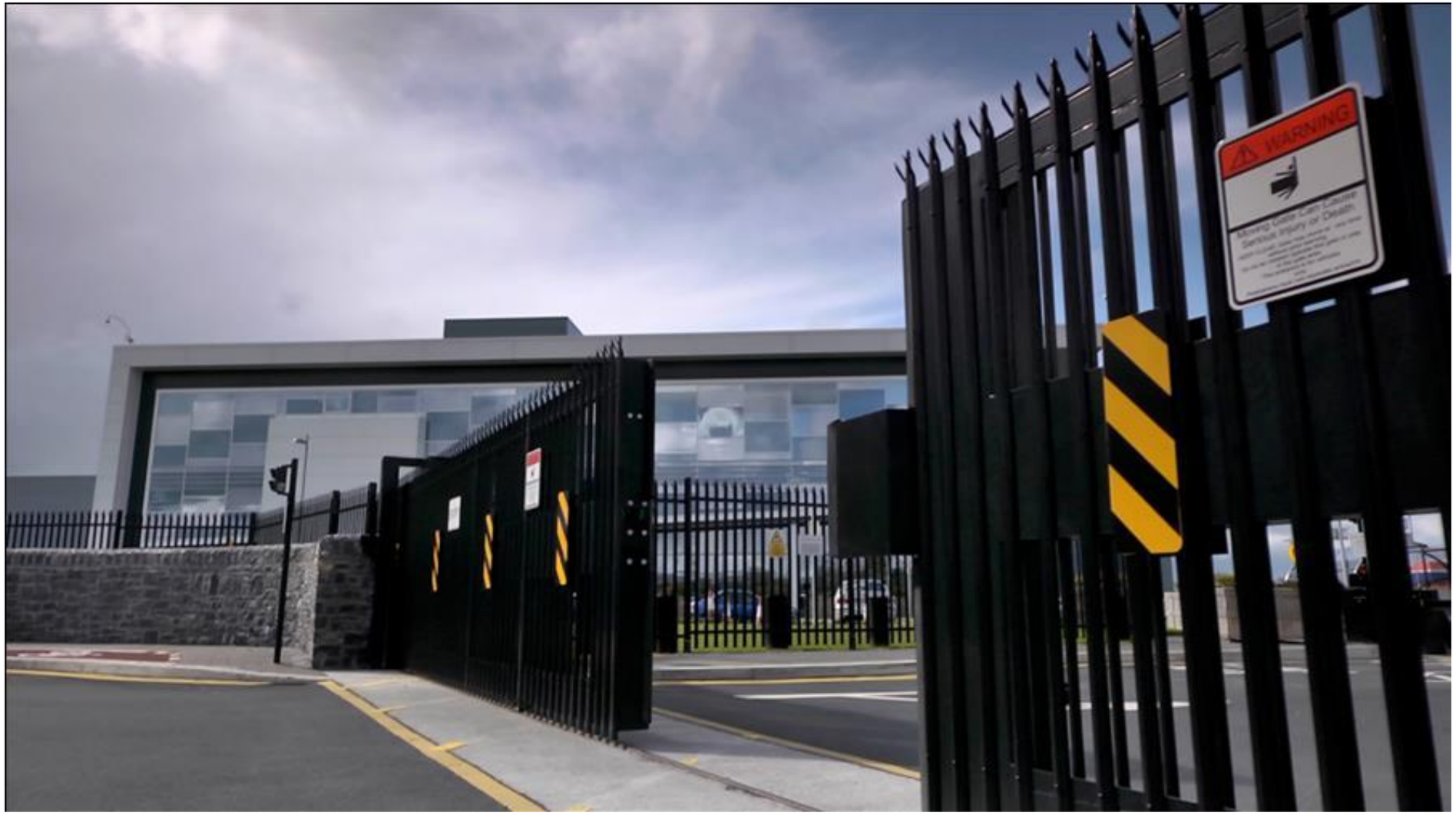
Update
MANAGEMENT

Anti-Virus/Anti-Malware
PROTECTION

Penetration **TESTING**

DDoS **DEFENSE**

Physical Security



Physical Security

Shared Responsibility (Misconfigurations)

OWASP Top 10 (2021) ~ IaaS, PaaS, Serverless

- A01: Broken Access Control
- A02: Cryptographic Failures
- A03: Injection
- A04: Insecure Design
- A05: Security Misconfiguration
- A06: Vulnerable and Outdated Components
- A07: Identification and Authentication Failures
- A08: Software and Data Integrity Failures
- A09: Security Logging and Monitoring Failures
- A10: Server-Side Request Forgery

Example: SQL Injection

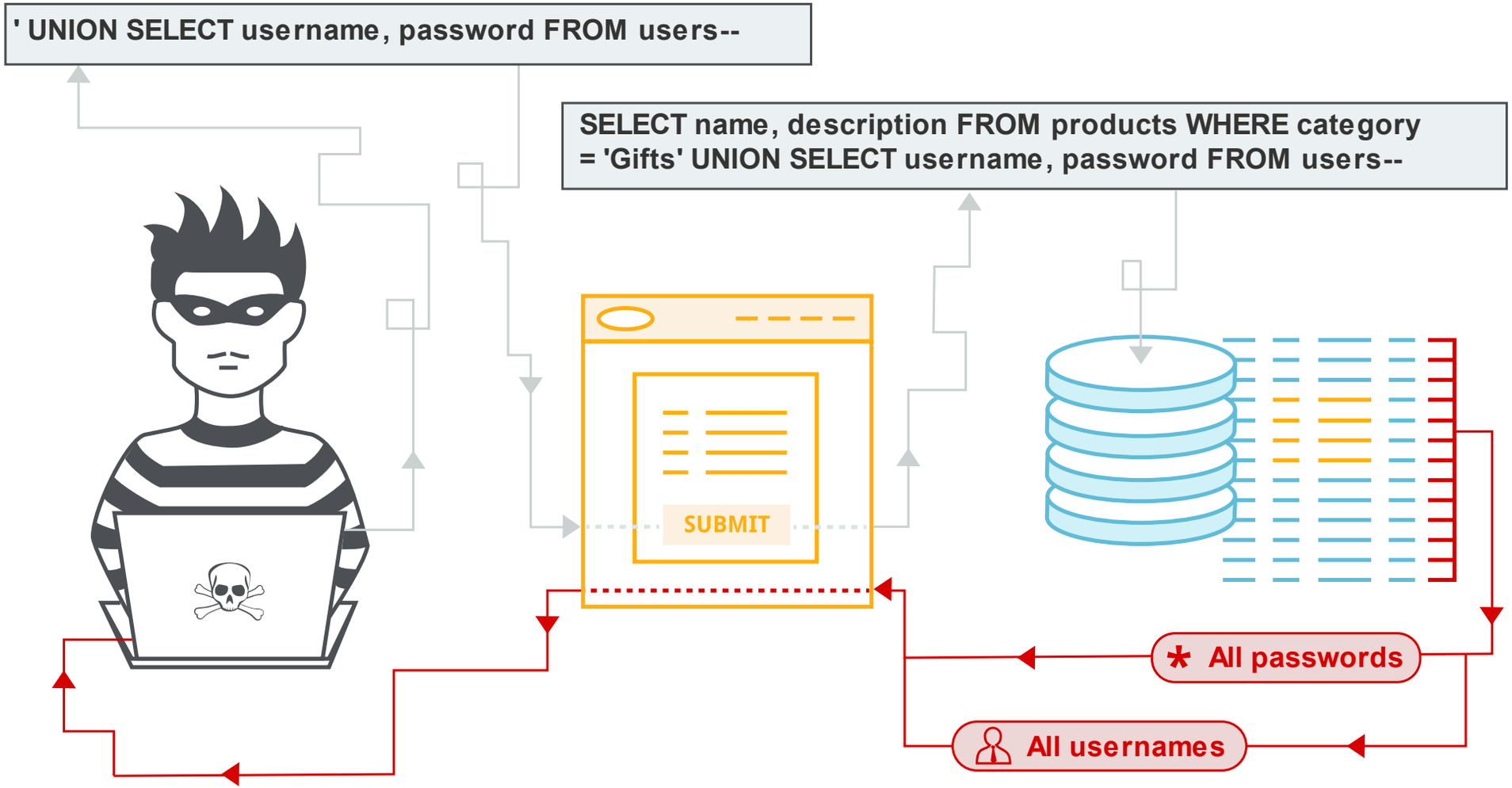
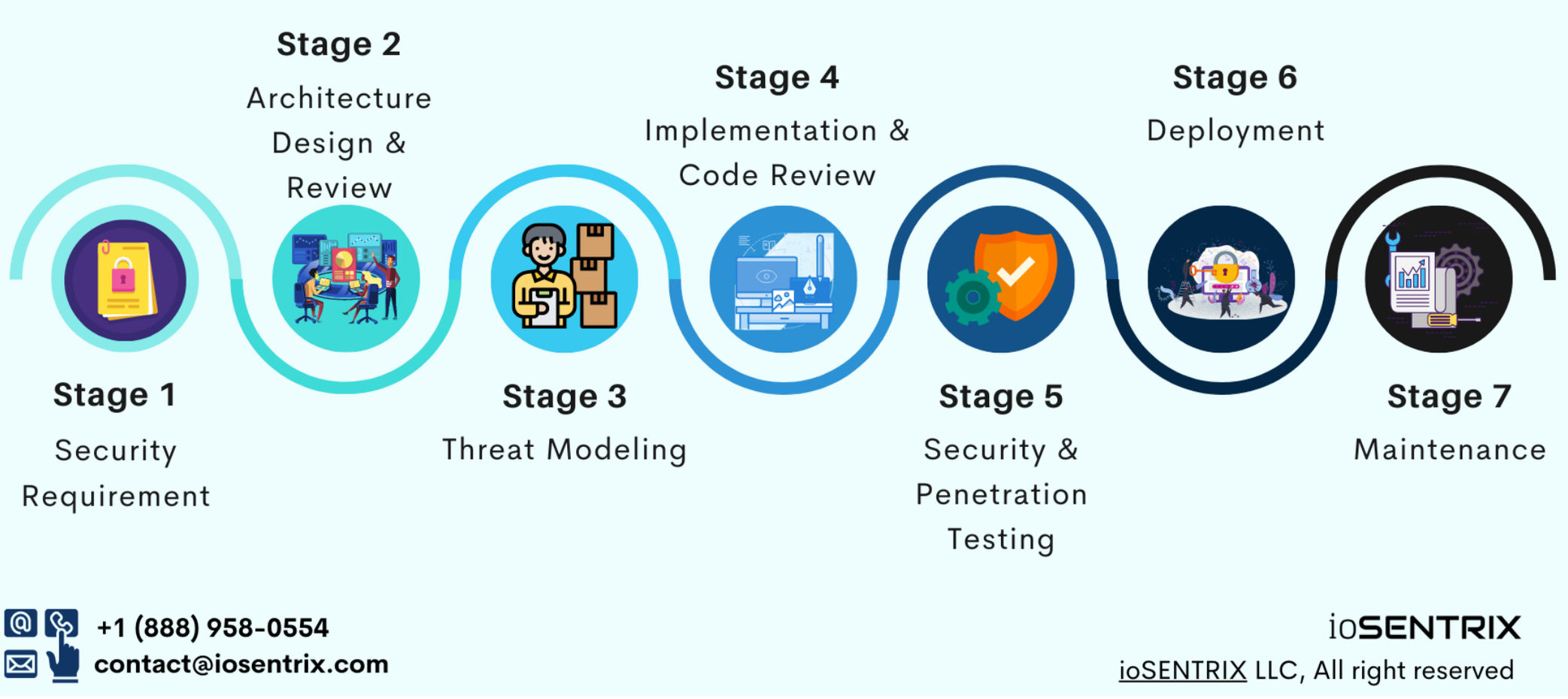


Image Source: https://portswigger.net/web-security/sql-injection

Secure SDLC



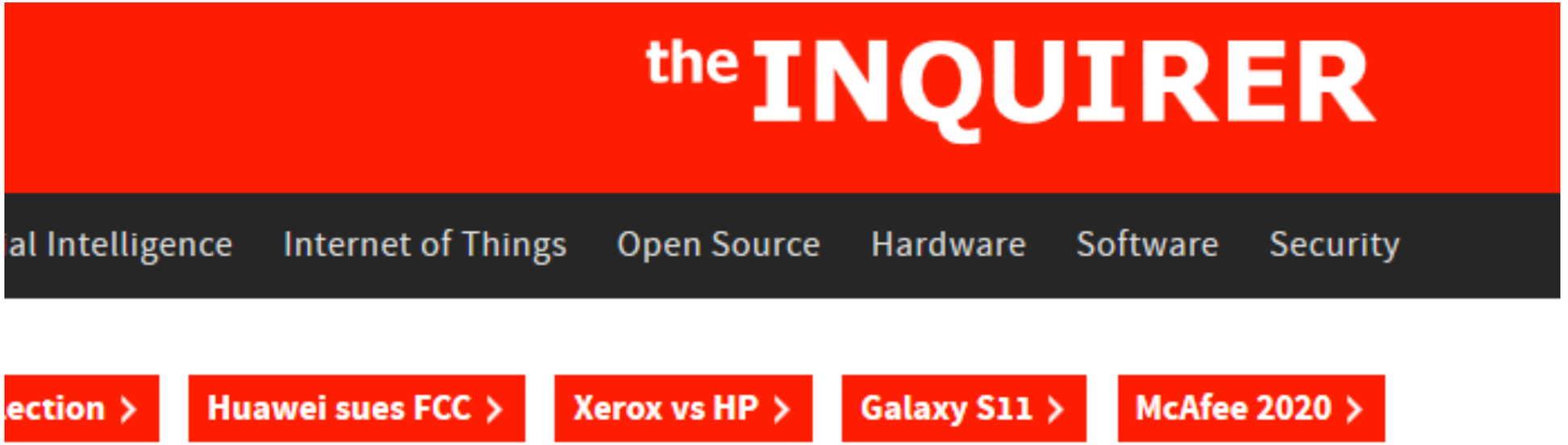
+1 (888) 958-0554
contact@iosentrix.com

ioSENTRIX

OWASP Cloud-Native Application Security Top 10 (2022)

- CNAS-1: Insecure cloud, container or orchestration configuration
  - Publicly open S3 bucket
  - Container runs as root
  - Container shares resources with the host (network interface, etc.)
  - Insecure Infrastructure-as-Code (IaC) configuration
- CNAS-2: Injection flaws (app layer, cloud events, cloud services)
  - SQL injection
  - XXE
  - NoSQL injection
  - OS command injection
  - Serverless event data injection

Default Credentials



PII in Public AWS S3 Buckets

SPIEGEL International

We Know Where You Parked

Massive Data Breach at VW Raises Questions about Vehicle Privacy

Already facing significant headwinds, VW has now been hit by a data protection nightmare. Location data from 800,000 electric vehicles and contact info from owners was accessible unprotected on the internet. And the company didn't even know about it.

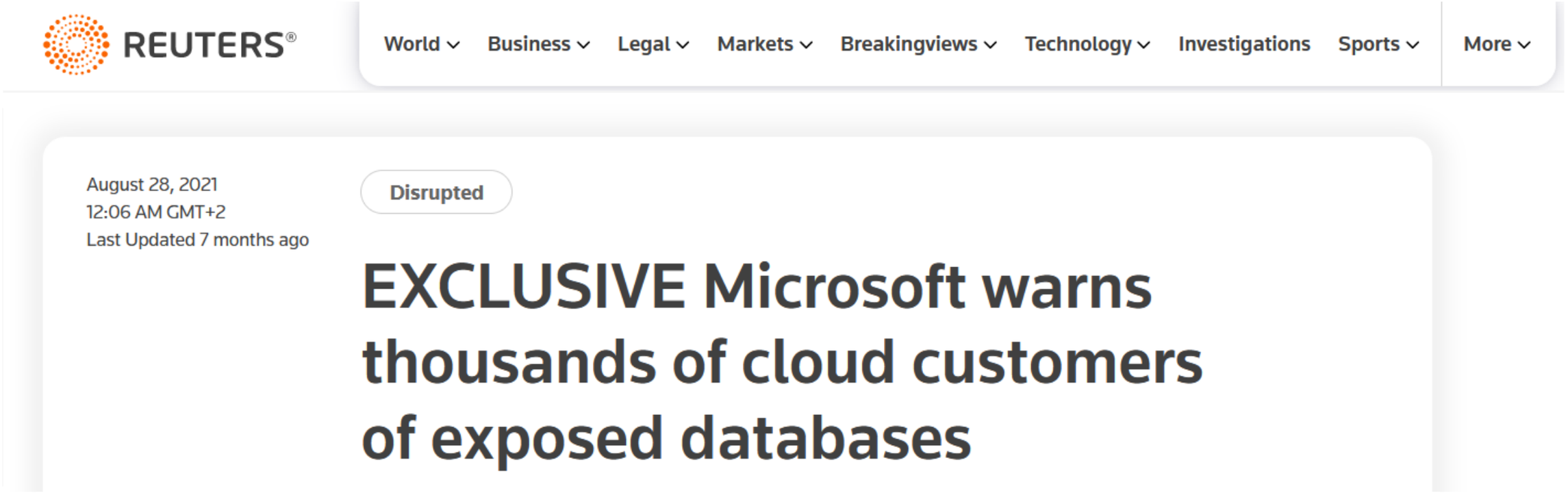
By **Patrick Beuth**, **Flüpke**, **Max Hoppenstedt**, **Michael Kreil**, **Marcel Rosenbach** und **Rina Wilkin**

03.01.2025, 16.19 Uhr

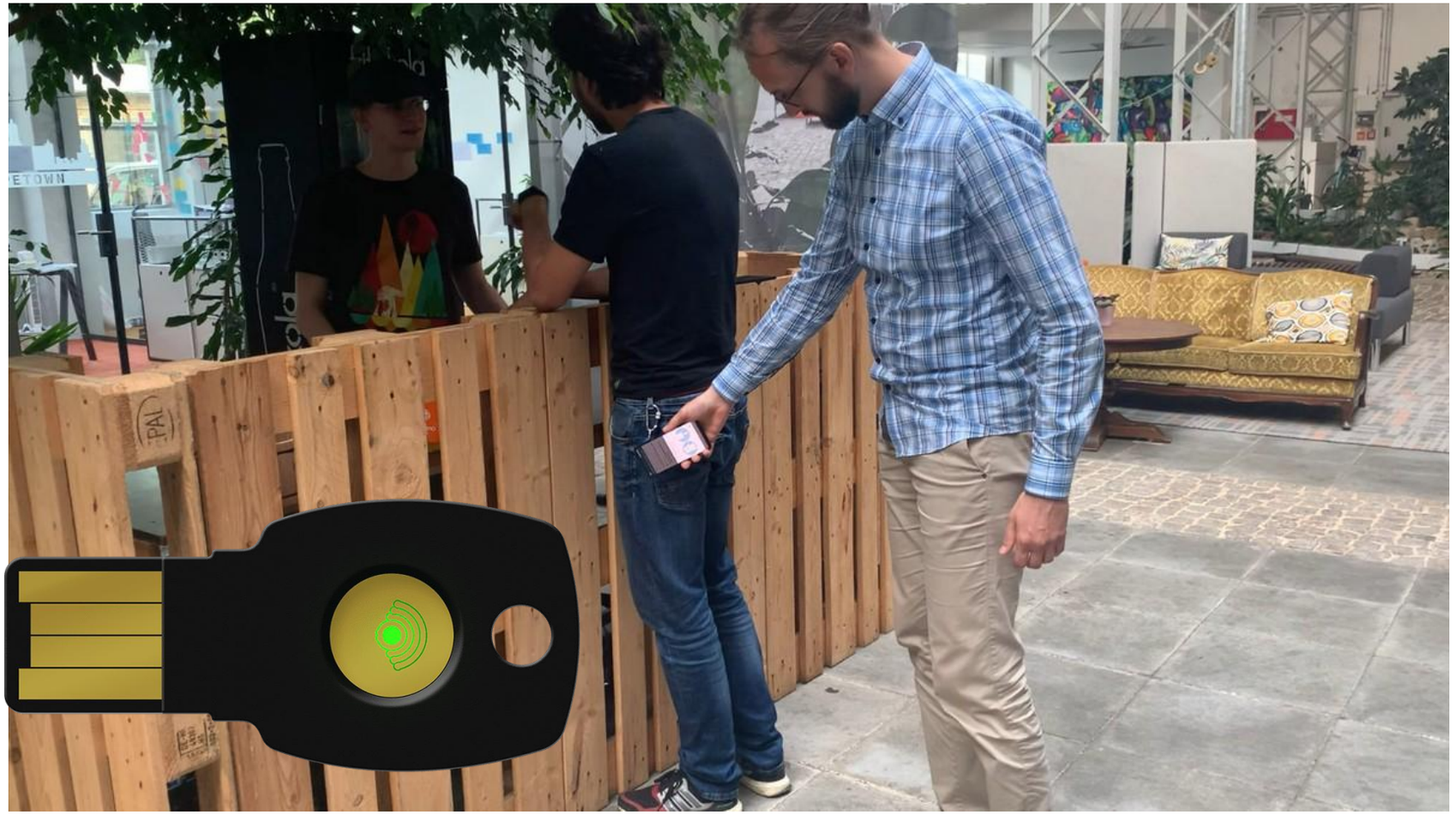
OWASP Cloud-Native Application Security Top 10 (2022)

- CNAS-3: Improper authentication & authorization
  - Unauthenticated API access on a microservice
  - Over-permissive cloud IAM role
  - Lack of orchestrator node trust rules (e.g. unauthorized hosts joining the cluster)
  - Unauthenticated orchestrator console access
  - Unauthrized or overly-permissive orchestrator access

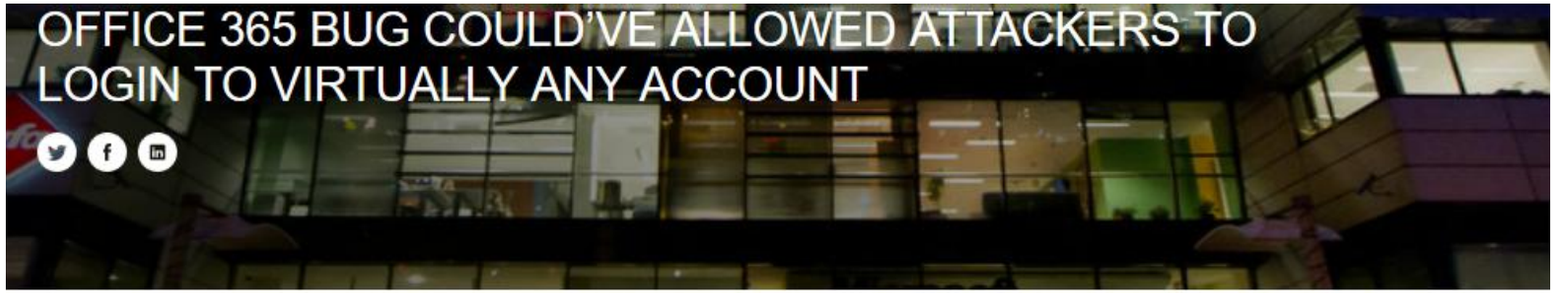
Sample Bug: Azure Cosmos DB



Sample Azure AD Bug:
PIN Bypass in Passkey Authentication



Sample Azure AD Bug: Missing UPN Suffix Validation in SAML



Security researchers in January discovered a critical vulnerability in the SAML implementation in Microsoft's Office 365 service that could allow an attacker to log in to a victim's account and gain full access to email, contacts, and other sensitive data.

The vulnerability was present in Office 365 for an unknown amount of time, and there is a long list of prominent companies that use the Web-based email and productivity suite, including Telefonika, British Telecom, Verizon, Microsoft, Cisco, Intel, and many others. The researchers who discovered the flaw said that an attacker targeting the vulnerability would be able to get to a wide range of important information on a victim's account.

RELATED POSTS

Google Adds New Anti-Phishing Feature to G Suite

Europol Dismantles International Fraud Ring

Phishing Attacks Using SSL Spike

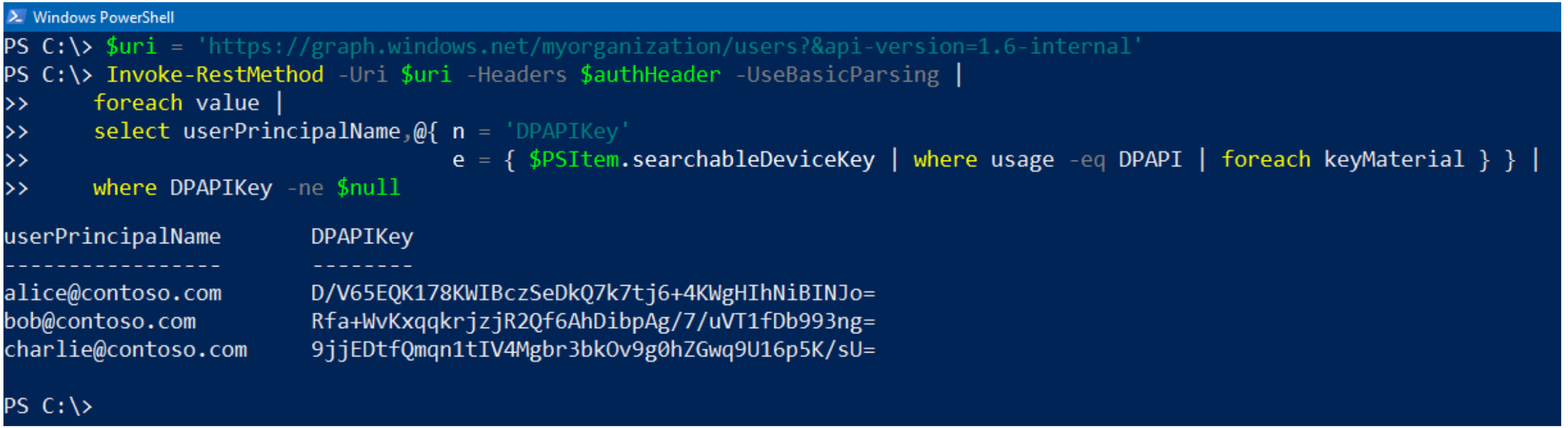
Apple to Switch Users to 2FA on iOS 11, macOS High Sierra

Preventing Forgery With Paper Fingerprinting

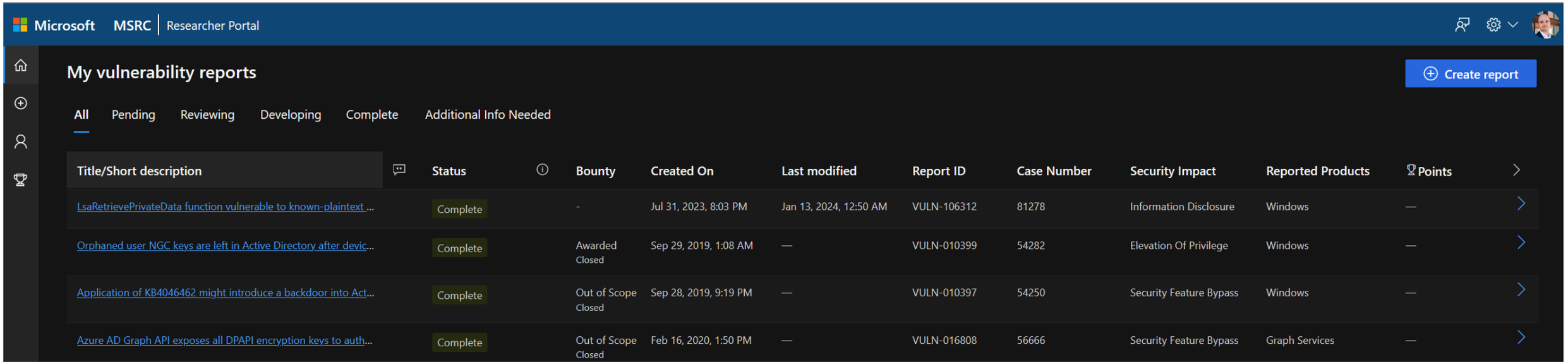
Sample Azure AD Bug:
Missing UPN Suffix Validation in SAML

```
<AttributeStatement>
    <Attribute Name="http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn">
        <AttributeValue>michael@dsinternals.com</AttributeValue>
    </Attribute>
</AttributeStatement>
<AuthnStatement AuthnInstant="2021-01-11T06:26:33.297Z">
    <AuthnContext>
        <AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:PasswordProt
```

Sample Azure AD Bug: Exposed DPAPI Encryption Keys



Reporting Bugs to MSRC



Microsoft Identity Bounty Program

Security Impact	Report Quality	Severity *Critical*	*Important*	*Moderate*	*Low*
Elevation of Privilege (*Involving Multi-factor Authentication Bypass*)	High Medium Low	$100,000 $75,000 $45,000	$50,000 $25,000 $10,000	$0	$0
Elevation of Privilege (*e.g Authentication Bypass or authentication flaw*)	High Medium Low	$40,000 $20,000 $8,000	$20,000 $10,000 $2,500	$0	$0
Spoofing (*e.g Cross-Site Scripting (XSS) or Cross-Site Request Forgery CSRF*))	High Medium Low	$20,000 $10,000 $6,000	$10,000 $5,000 $1,500	$0	$0
Information Disclosure (*e.g. Sensitive Data Exposure*)	High Medium Low	$12,000 $6,000 $4,500	$7,500 $3,000 $1,500	$0	$0
Standards Design Vulnerabilities (*Some limitations apply, see Out of Scope section below*)	High Medium Low	$100,000 $60,000 $25,000	$30,000 $20,000 $2,500	$0	$0
Standards-based implementation vulnerabilities (*Some limitations apply, see Out of Scope section below*)	High Medium Low	$75,000 $50,000 $20,000	$25,000 $10,000 $2,500	$0	$0

OWASP Cloud-Native Application Security Top 10 (2022)

- CNAS-4: CI/CD pipeline & software supply chain flaws
  - Insufficient authentication on CI/CD pipeline systems
  - Use of untrusted images
  - Use of stale images
  - Insecure communication channels to registries
  - Overly-permissive registry access
  - Using a single environment to run CI/CD tasks for projects requiring different levels of security
- CNAS-5: Insecure secrets storage
  - Orchestrator secrets stored unencrypted
  - API keys or passwords stored unencrypted inside containers
  - Hardcoded application secrets
  - Poorly encrypted secrets (e.g., use of obsolete encryption methods, use of encoding instead of encryption)
  - Mounting of storage containing sensitive information

Passwords in Public AWS S3 Buckets

threatpost Cloud Security / Malware / Vulnerabilities / InfoSec Insider / Podcasts

Verizon Wireless Internal Credentials, Infrastructure Details Exposed in Amazon S3 Bucket

Secrets in Source Code Repositories

```
def ftp_file():
    ftp_host='[redacted]
    ftp_user='[redacted]
    ftp_passwo[redacted]
    ftp_destination='/Test/test/'
    source_file_name='/home/airflow/airflow/dags/ftpupload/weeklyrefresh.txt'

    with ftputil.FTPHost(ftp_host, ftp_user, ftp_password) as ftp_host:

        buf_size = 4 * 1024
        print("FTP Host: " + str(ftp_host))
        with open(source_file_name, 'rb') as source_file:
#           print("Opened source file: " + source_file.name)
#           ftp_host.makedirs(args.ftp_destination)
            print("Made FTP directory: " + ftp_destination)
            target_file_name = ftp_destination + get_name(source_file_name)
            with ftp_host.open(target_file_name, "wb") as target_file:
#               print("Opened target file: " + str(target_file_name))
                while True:
                    data=source_file.read(buf_size)
                    if not data: break
                    target_file.write(data)
                print("Copy complete: "    + source_file_name)
```

Hardware Security Module (HSM)

OWASP Cloud-Native Application Security Top 10 (2022)

- CNAS-6: Over-permissive or insecure network policies
  - Over-permissive pod to pod communication allowed
  - Internal microservices exposed to the public Internet
  - No network segmentation defined
  - End-to-end communications not encrypted
  - Network traffic to unknown or potentially malicious domains not monitored and blocked
- CNAS-7: Using components with known vulnerabilities
  - Vulnerable 3rd party open-source packages
  - Vulnerable versions of application components
  - Use of known vulnerable container images

Public RDP Access

welivesecurity™ BY ESET®

First BlueKeep attacks prompt fresh warnings

The infamous vulnerability has been exploited for a cryptocurrency mining campaign, but more damaging attacks may still be in store

Amer Owaida 11 Nov 2019 - 05:16PM

Bugs in Applications

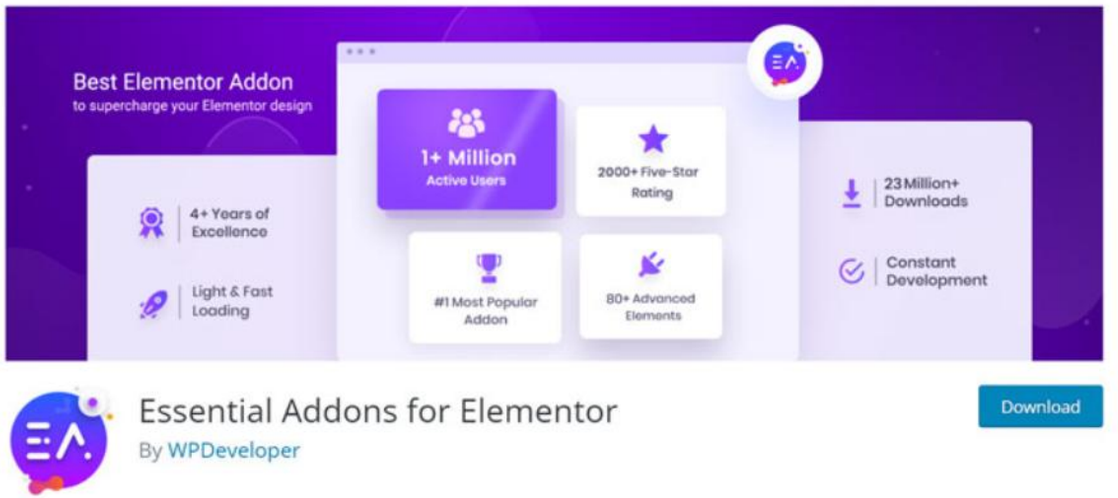
The Hacker News

Subscribe to Newsletter

Home | Data Breaches | Cyber Attacks | Vulnerabilities | Malware | Offers | Contact

Critical Bug Found in WordPress Plugin for Elementor with Over a Million Installations

February 01, 2022 Ravie Lakshmanan



Popular This Week

Chinese Hackers Target VMware Horizon Servers with Log4Shell to Deploy Rootkit

CISA Warns of Active Exploitation of Critical Spring4Shell Vulnerability

Apple Issues Patches for 2 Actively Exploited Zero-Days in iPhone, iPad and Mac Devices

OWASP Cloud-Native Application Security Top 10 (2022)

- CNAS-8: Improper assets management
  - Undocumented microservices & APIs
  - Obsolete & unmanaged cloud resources
- CNAS-9: Inadequate ‘compute’ resource quota limits
  - Resource-unbound containers
  - Over-permissive request quota set on APIs
- CNAS-10: Ineffective logging & monitoring (e.g. runtime activity)
  - No container or host process activity monitoring
  - No network communications monitoring among microservices
  - No resource consumption monitoring to ensure availability of critical resources
  - Lack of monitoring on orchestration configuration propagation and stale configs

Subdomain Takeovers

BLEEPINGCOMPUTER

NEWS ▾ DOWNLOADS ▾ VIRUS REMOVAL GUIDES ▾ TUTORIALS ▾ DEALS ▾

Home › News › Security › Starbucks Abandons Azure Site, Exposed Subdomain to Hijacking

Starbucks Abandons Azure Site, Exposed Subdomain to Hijacking

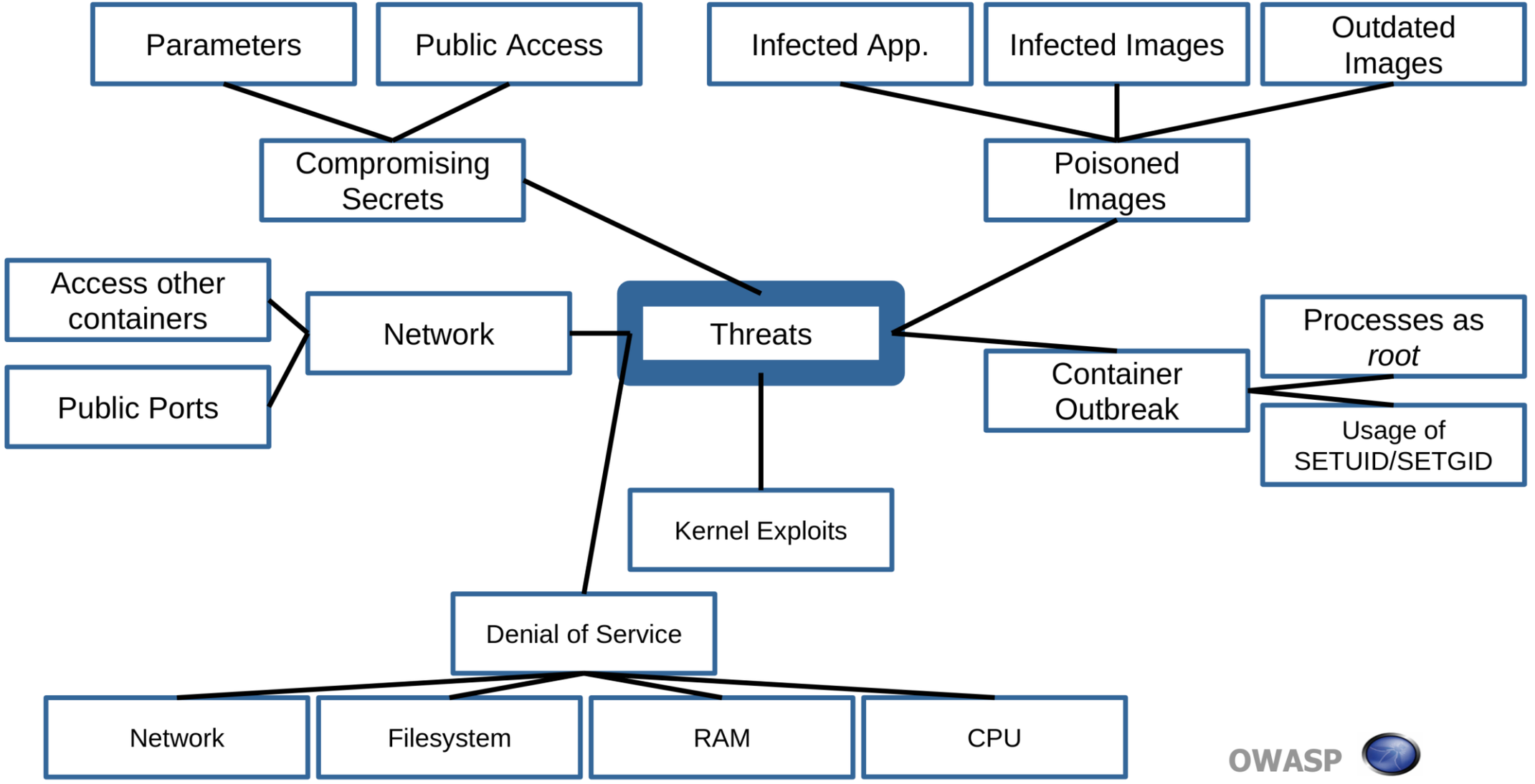
By Ionut Ilascu

August 29, 2019 05:46 AM 2

OWASP Docker Top 10

- D01 - Secure User Mapping
- D02 - Patch Management Strategy
- D03 - Network Segmentation and Firewalling
- D04 - Secure Defaults and Hardening
- D05 - Maintain Security Contexts
- D06 - Protect Secrets
- D07 - Resource Protection
- D08 - Container Image Integrity and Origin
- D09 - Follow Immutable Paradigm
- D10 - Logging

OWASP: Docker Threat Modeling



OWASP

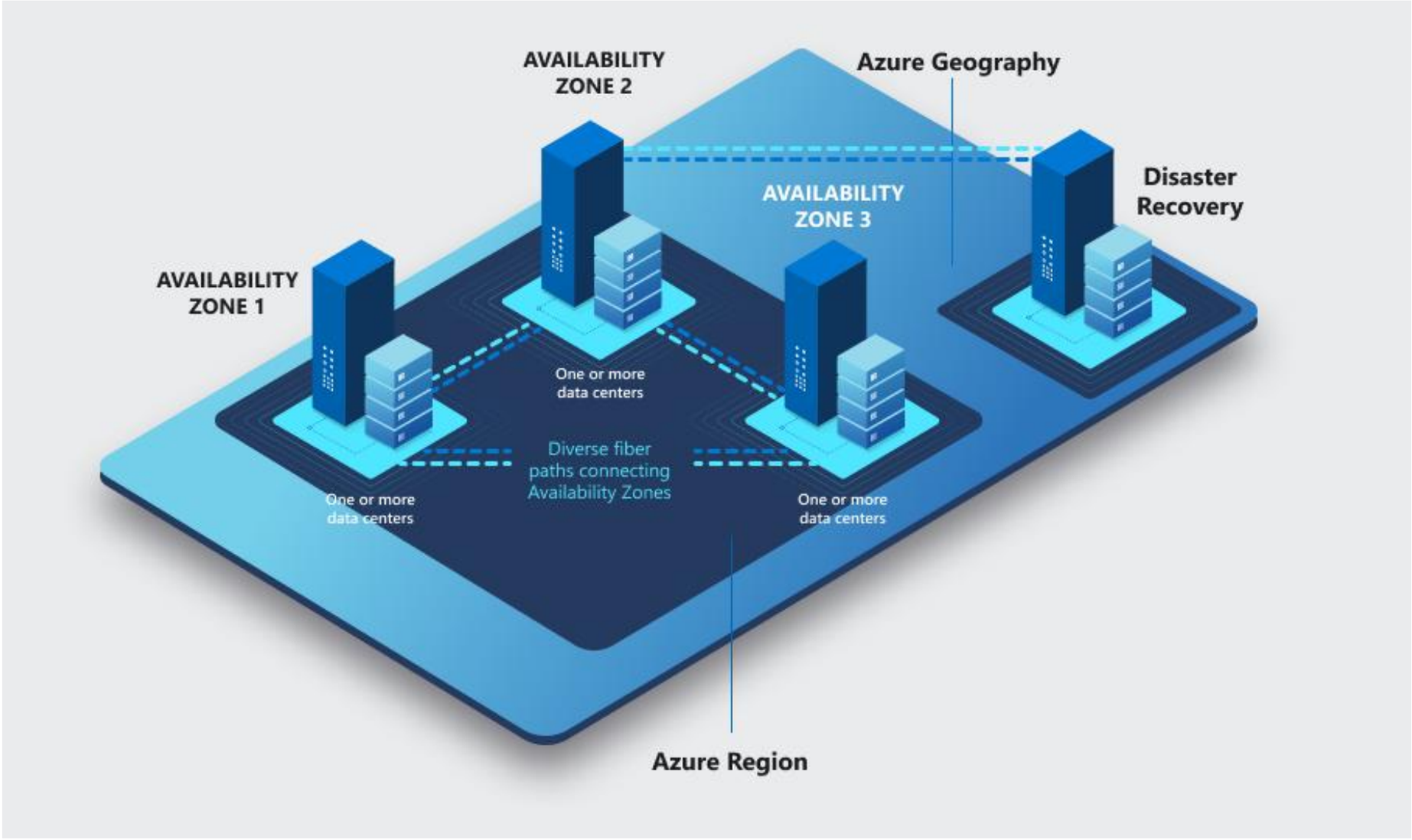
OWASP Top 10 Low-Code/No-Code Security Risks

- Risk #1: Privilege Escalation
- Risk #2: Insecure Authentication
- Risk #3: Data Leakage
- Risk #4: Oversharing
- Risk #5: Data and Secret Handling Failures
- Risk #6: Misconfiguration
- Risk #7: Business Continuity, Resiliency and Ownership Failures
- Risk #8: App impersonation
- Risk #9: Dependency Injection
- Risk #10: Unmanaged Custom Code

OWASP TOP 10 Cloud Security Risks

- R1: Accountability and Data Ownership
- R2: User Identity Federation
- R3: Regulatory Compliance
- R4: Business Continuity and Resiliency
- R5: User Privacy and Secondary Usage of Data
- R6: Service and Data Integration
- R7: Multi Tenancy and Physical Security
- R8: Incident Analysis and Forensic Support
- R9: Infrastructure Security
- R10: Non-Production Environment Exposure

Business Continuity / Disaster Recovery



Denial of Service

Long-Term Viability, Vendor Lock-In, Cloud Exit Strategy

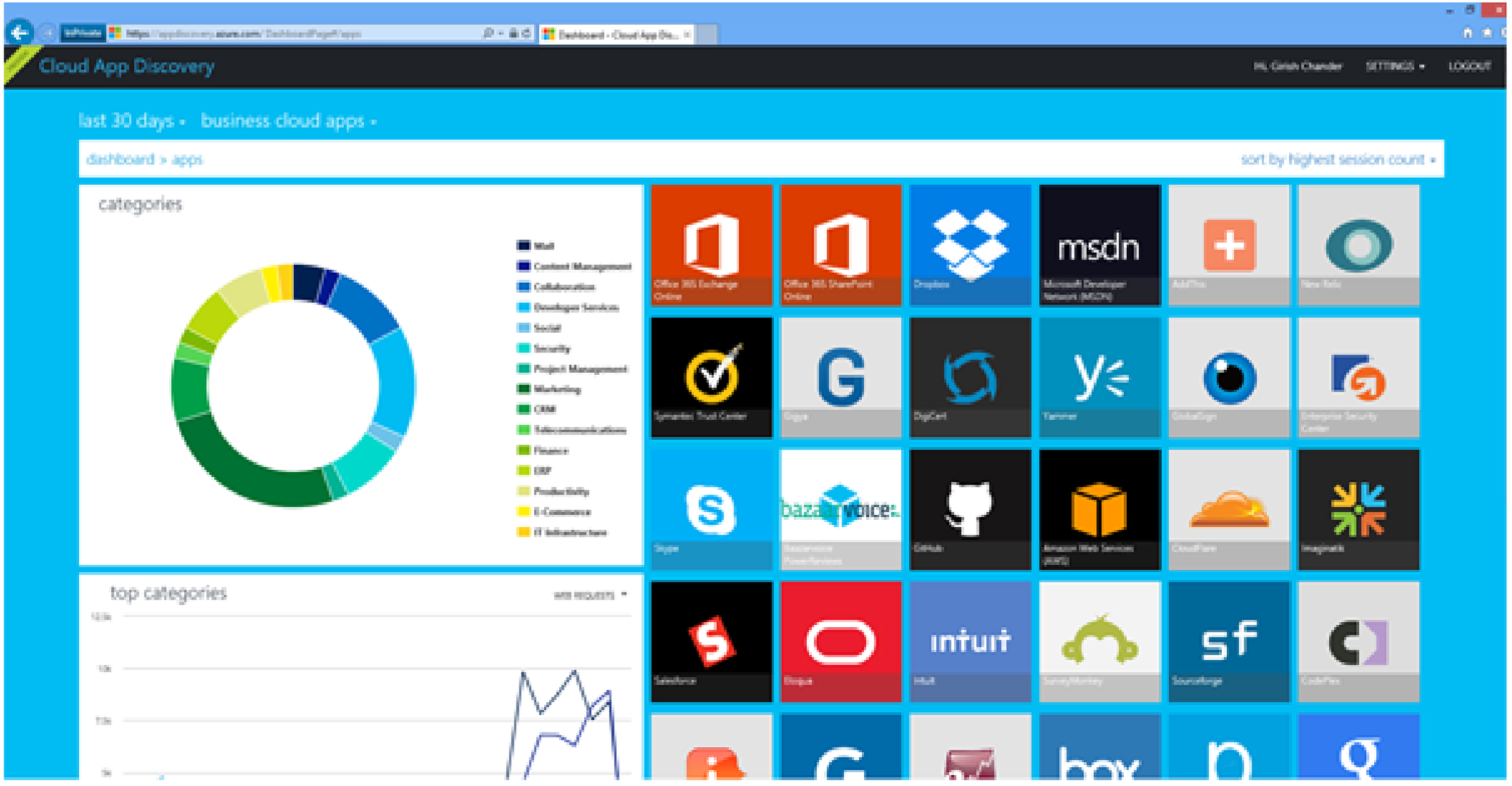
Novinky.cz

Český Jablotron zablokoval datové služby pro klienty v Rusku a Bělorusku

2. 3. 2022, 7:30 – Jablonec nad Nisou – ČTK

Český výrobce zabezpečovacích zařízení Jablotron zablokoval datové služby pro uživatele v Rusku a Bělorusku. Chce tak burcovat ruskou veřejnost proti prezidentovi Vladimiru Putinovi a jeho lidem. Kromě zastavení prodeje do Ruska a Běloruska zablokuje Jablotron datové služby pro své už dříve namontované výrobky v těchto zemích. Uvedl to zakladatel Jablotronu Dalibor Dědek.

Shadow IT – Anyone can use DropBox, GitHub,etc.



Authentication in Cloud Applications

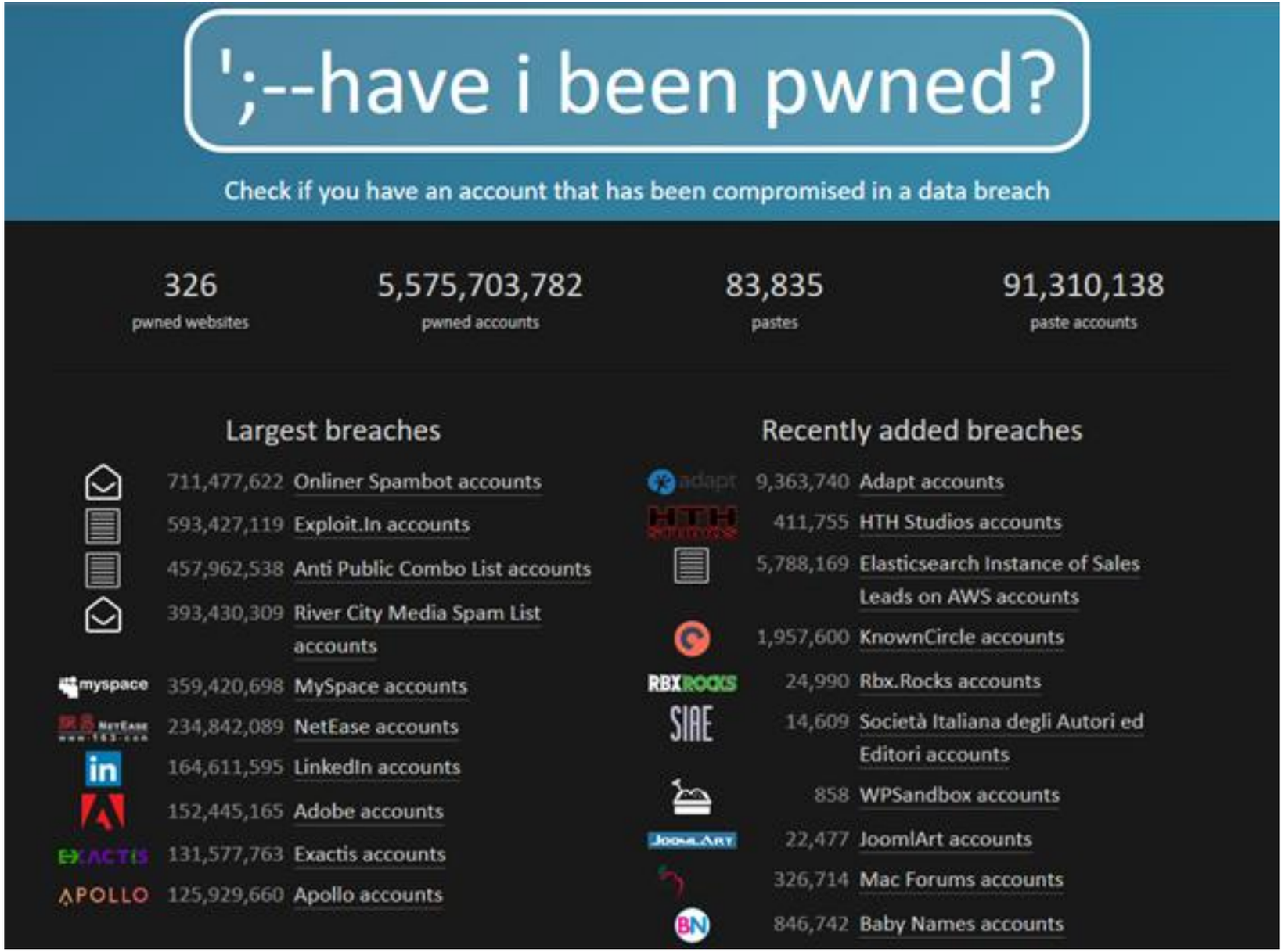
Authentication

- Basic Concepts
  - Two-factor authentication
  - OTPs
- Identity Federation
  - OAuth
  - OpenID Connect
  - SAML

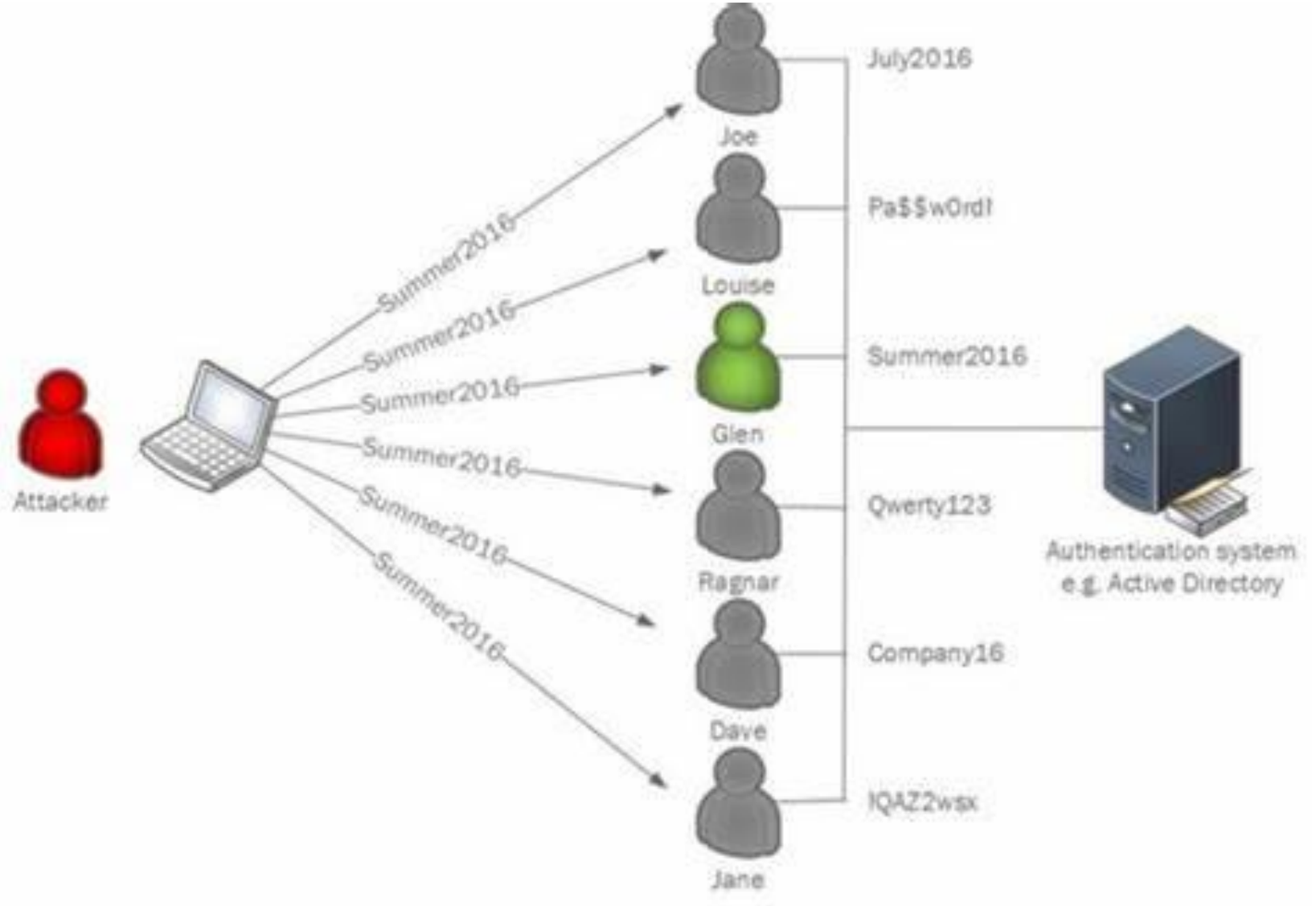
Passwords Are Dead

- Dictionary Attacks
- Password Leaks
- Phishing
- Keyloggers
- Shoulder Surfing
- …

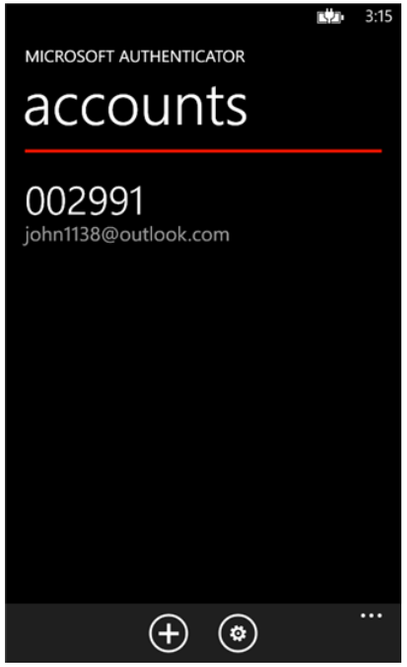
Passwords Are Dead



Password Spraying



Multi-Factor Authentication



One-time Passwords

- HMAC-based One-time Password (HOTP)
- Time-based One-time Password (TOTP)
- Out-of-Band Transaction Authentication Numbers (TANs)
  - Pre-generated (POTP)
  - SMS
  - Push Notifications

TOTP

$$HMAC(K, C) := SHA1(K \oplus 0x5C5C... \| SHA1(K \oplus 0x3636... \| C))$$

$$HOTP(K, C) := Truncate(HMAC(K,C)) \ \& \ 0x7FFFFFFF$$

K – Secret Key
C – Counter/Clock

MFA Method Strength

Microsoft Authenticator
Passwordless

Password
+ Microsoft Authenticator
Number match

Password
+ Hardware Tokens OTP

Password
+ Software Tokens OTP

Password
+ Voice

Password
+ SMS

Phishing resistant

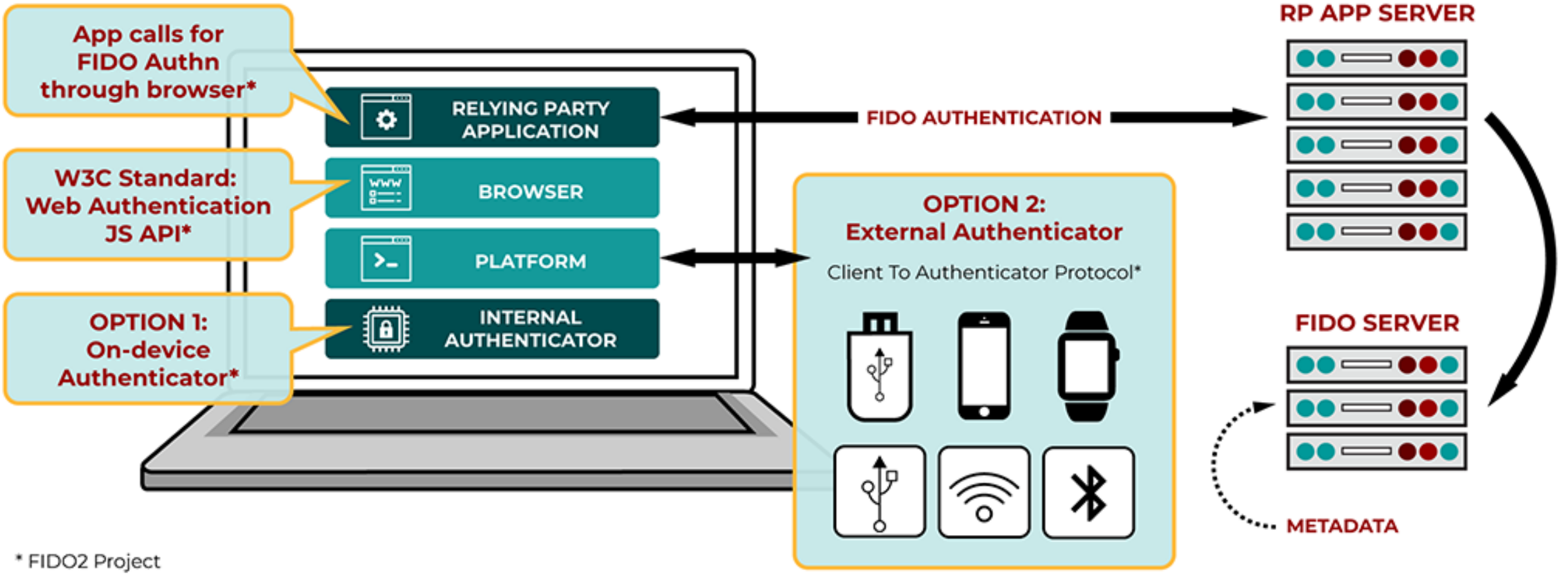
Certificate based
authentication

FIDO2 security keys

Windows Hello
for Business

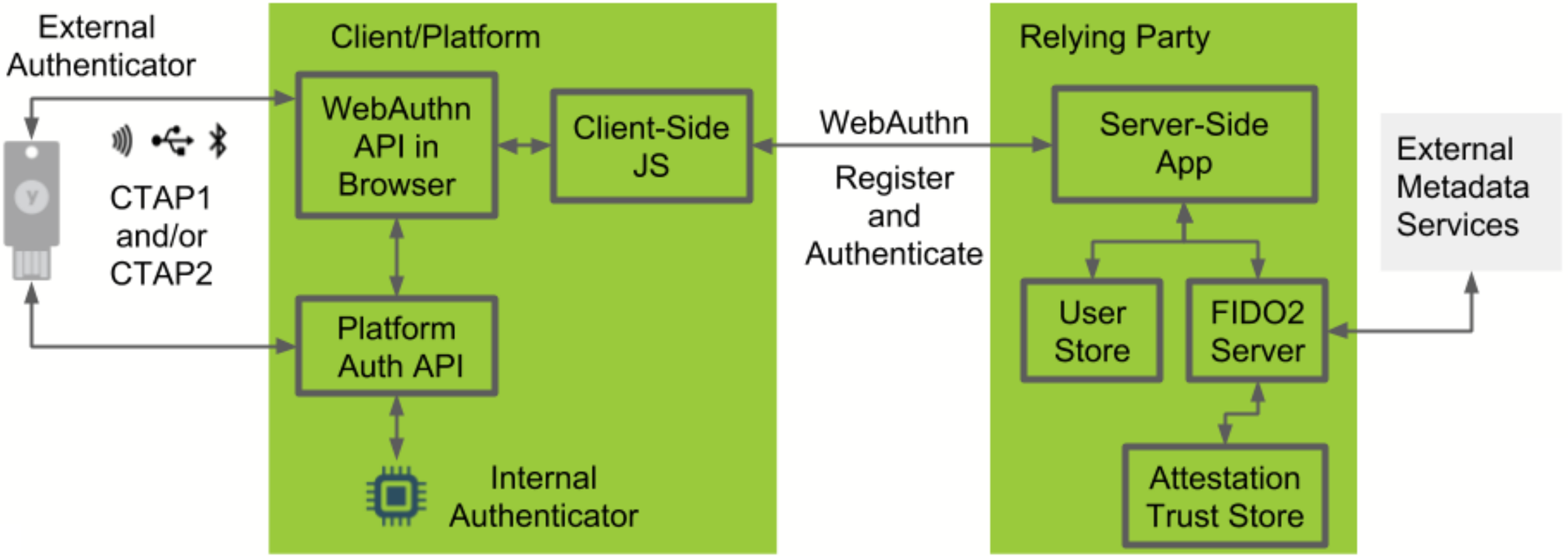
Passkey in Microsoft Authenticator
(device-bound)

Passkeys / FIDO2 / W3C Web Authentication



* FIDO2 Project

Passkey Protocols



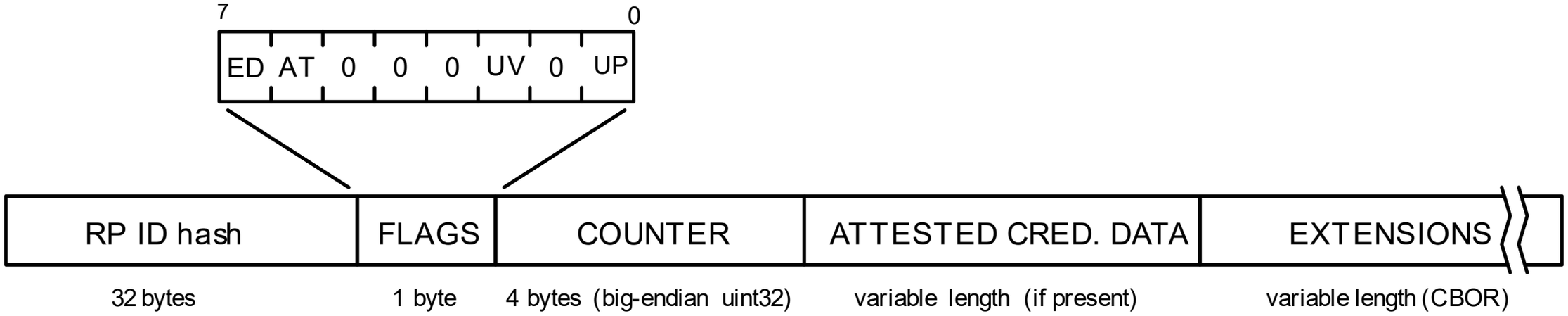
WebAuthn API

```
// Detection of WebAuthn support
if (!window.PublicKeyCredential) { /* Client not capable. Handle error. */ }

// Registration
const credential = await navigator.credentials.create({
    publicKey: publicKeyCredentialCreationOptions
});

// Assertion (Logon)
const credential = await navigator.credentials.get({
    publicKey: publicKeyCredentialRequestOptions
});
```

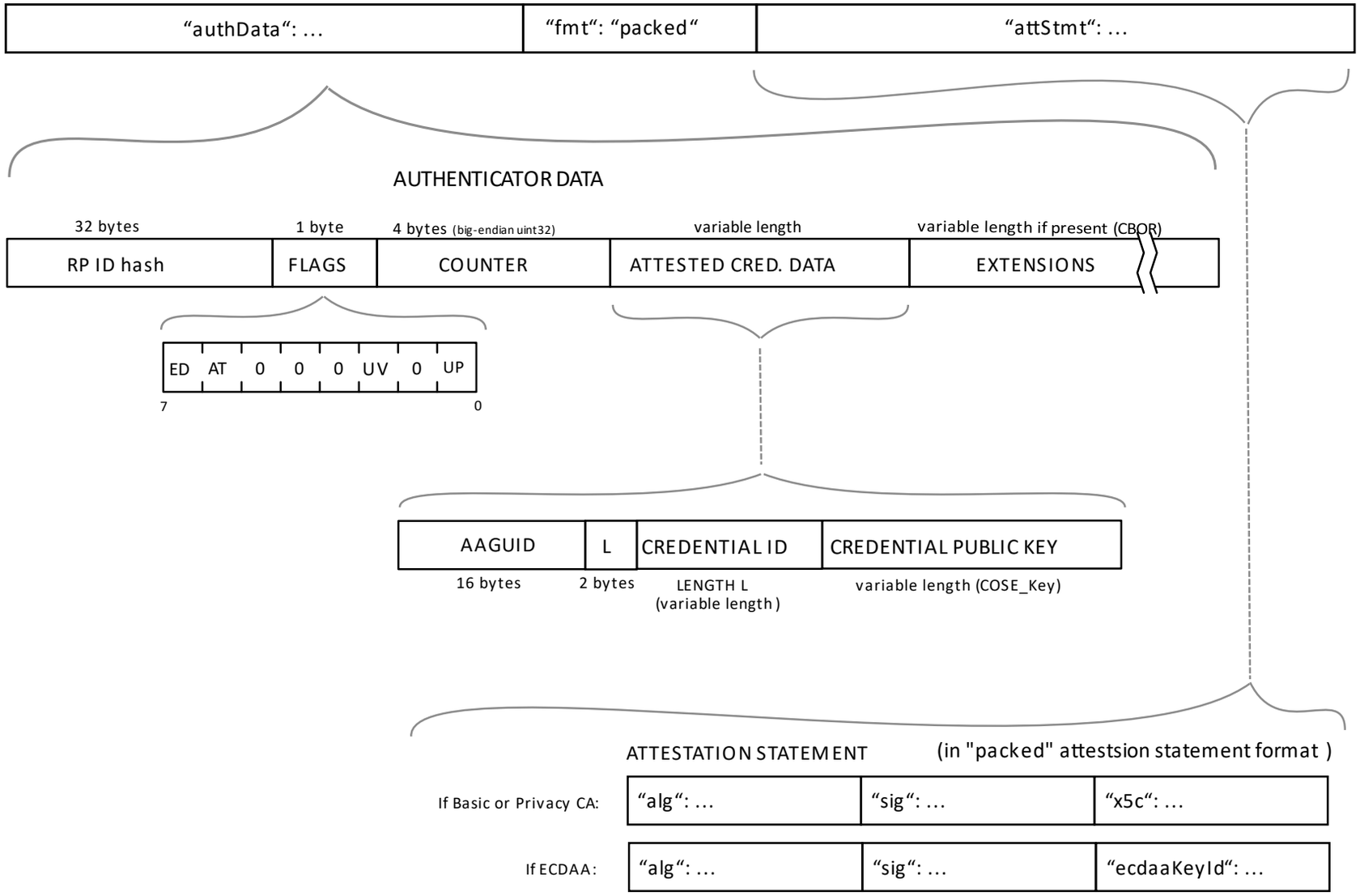
Public Key Credential Creation Options

```
const publicKeyCredentialCreationOptions = {
    challenge: Uint8Array.from(randomStringFromServer, c => c.charCodeAt(0)),
    rp: {
        name: "Microsoft",
        id: "login.microsoft.com",
    },
    user: {
        id: Uint8Array.from(window.atob("MIIBkzCCATigAwIBAjCCAZMwggE4oAMCAQIwggGTMII="), c=>c.charCodeAt(0)),
        name: "michael@dsinternals.com",
        displayName: "Michael Grafnetter",
    },
    pubKeyCredParams: [{alg: -7, type: "public-key"}],
    authenticatorSelection: {
        authenticatorAttachment: "cross-platform",
        requireResidentKey: true,
        userVerification: "required"
    },
    timeout: 60000,
    attestation: "direct"
};
```

Authenticator Data



Attestation Object



Language-Specific SDKs (webauthn.io)

Go
duo-labs/webauthn
Duo Labs
Server

Go
duo-labs/webauthn.io
Duo Labs
Demo

Go
koesie10/webauthn
Koen Vlaswinkel
Server

Java
duo-labs/android-webauthn-authenticator
Duo Labs
Authenticator

Java
google/webauthndemo
Google
Demo

Java
webauthn4j/webauthn4j
Yoshikazu Nojima
Server

Java
Yubico/java-webauthn-server
Yubico
Server

Javascript
fido-alliance/webauthn-demo
Fido Alliance
Demo

.NET
abergs/fido2-net-lib
Anders Åberg
Server/Demo

Python
duo-labs/py_webauthn
Duo Labs
Server/Demo

Ruby
cedarcode/webauthn-ruby
Cedarcode
Server

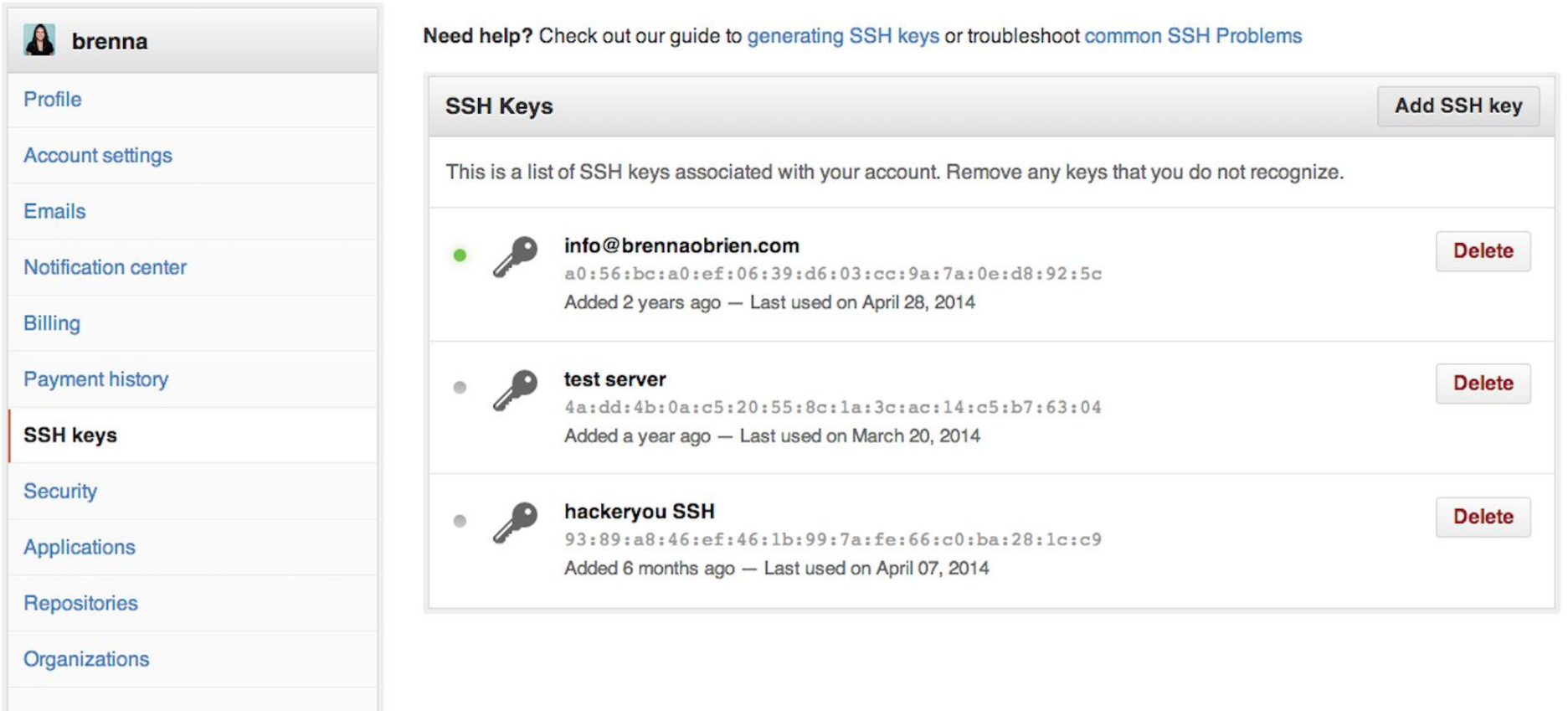
Chrome Extension
google/virtual-authenticators-tab
Nina Satragno
Authenticator

Problem: Lost Device

- Multiple Devices
- Recovery Questions
  What is the maiden name of your mother? 
- E-Mail Verification 
- POTP
- In-Person Verification (used in enterprise environments, but not usable with cloud services)

Application-Specific Passwords

Some protocols (or their implementations) like SSH, SFTP or IMAP do not support 2FA



Claims-Based Identity

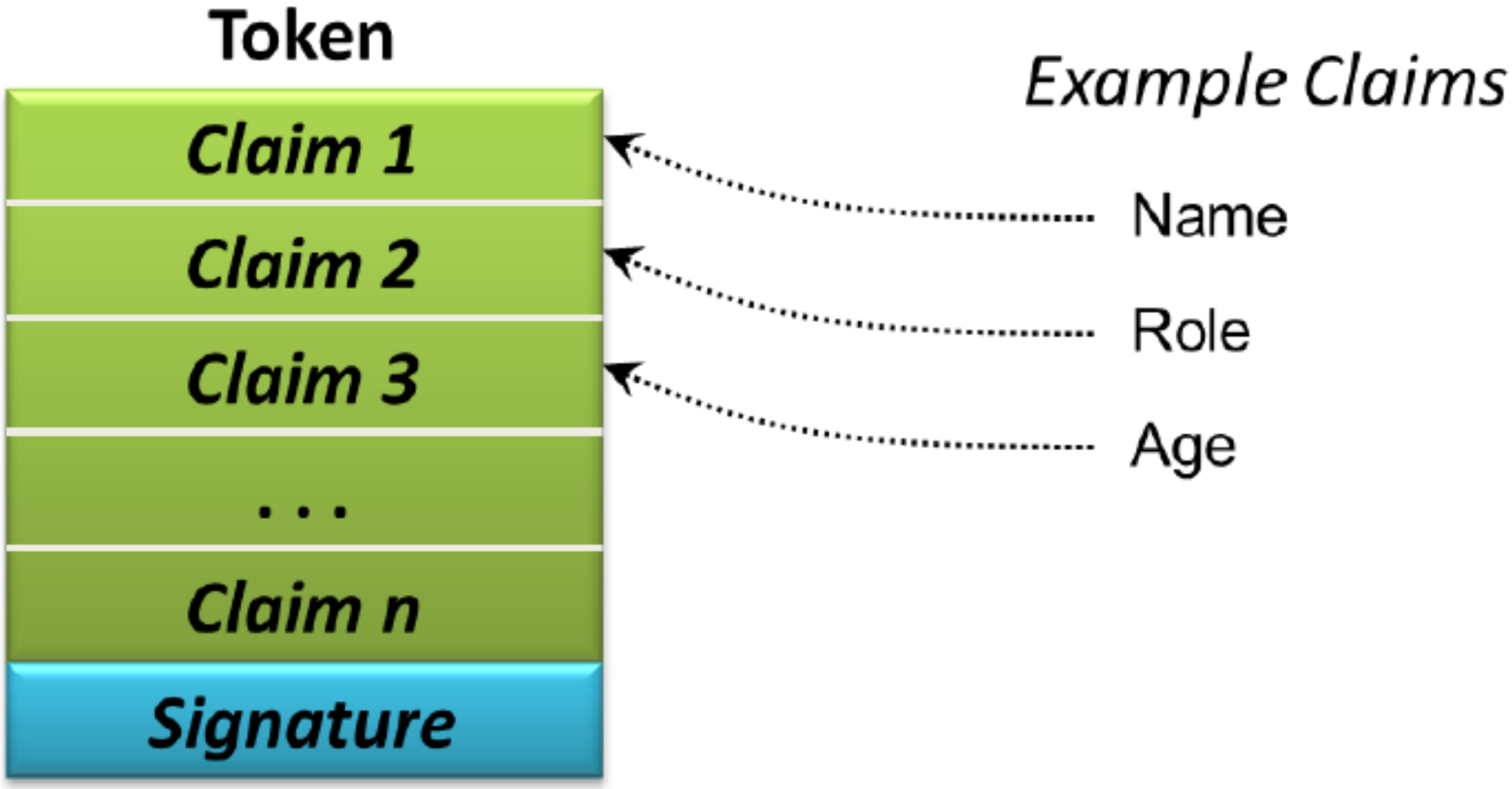
Most Common Technologies

- OAuth
- OpenID Connect
- SAML
- WS-*
- JWT

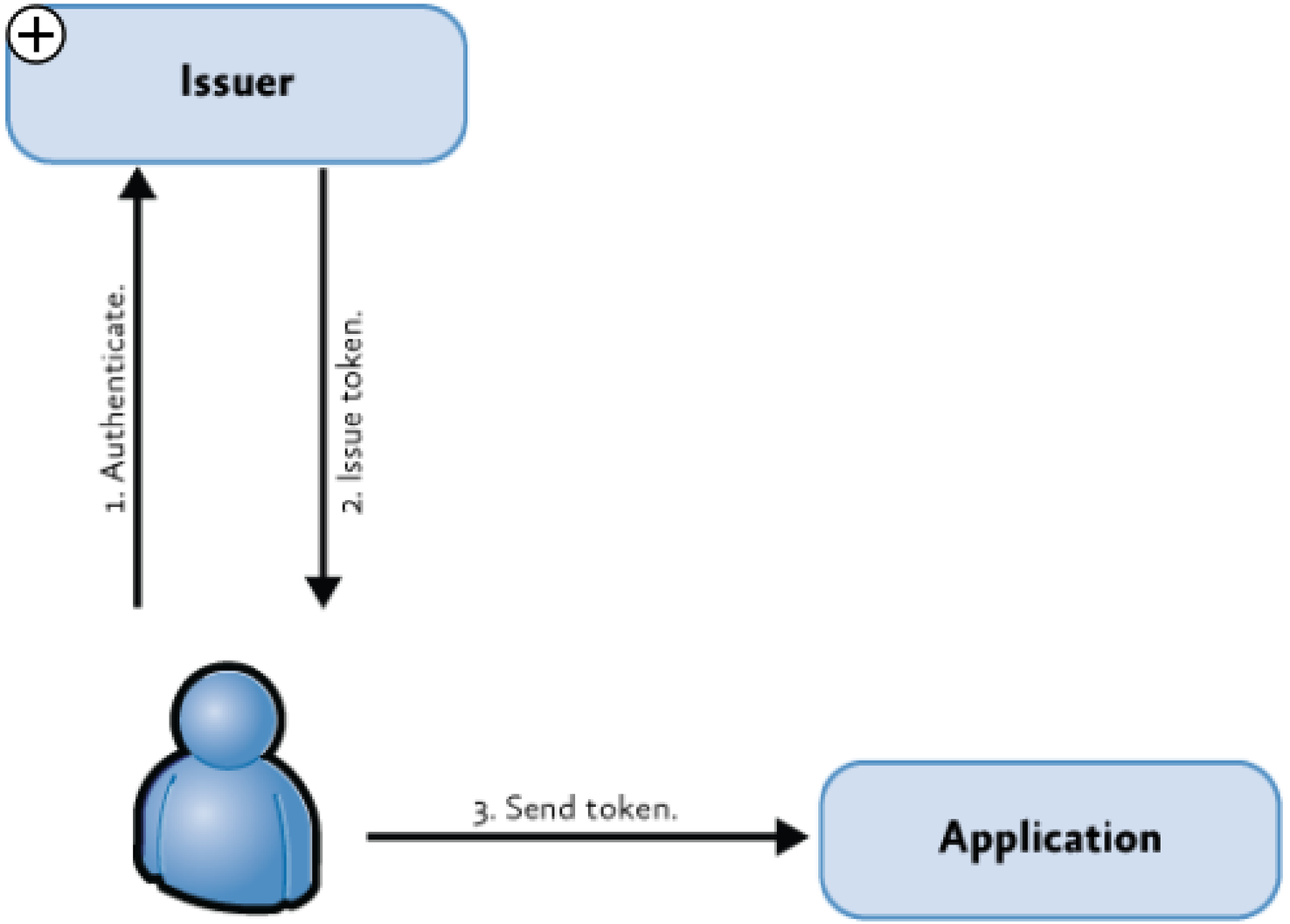
Claims-based Identity

- First Name: John
- Last Name: Doe
- Login: John
- Mail: john@doe.com
- Role: User
- Role: Administrator

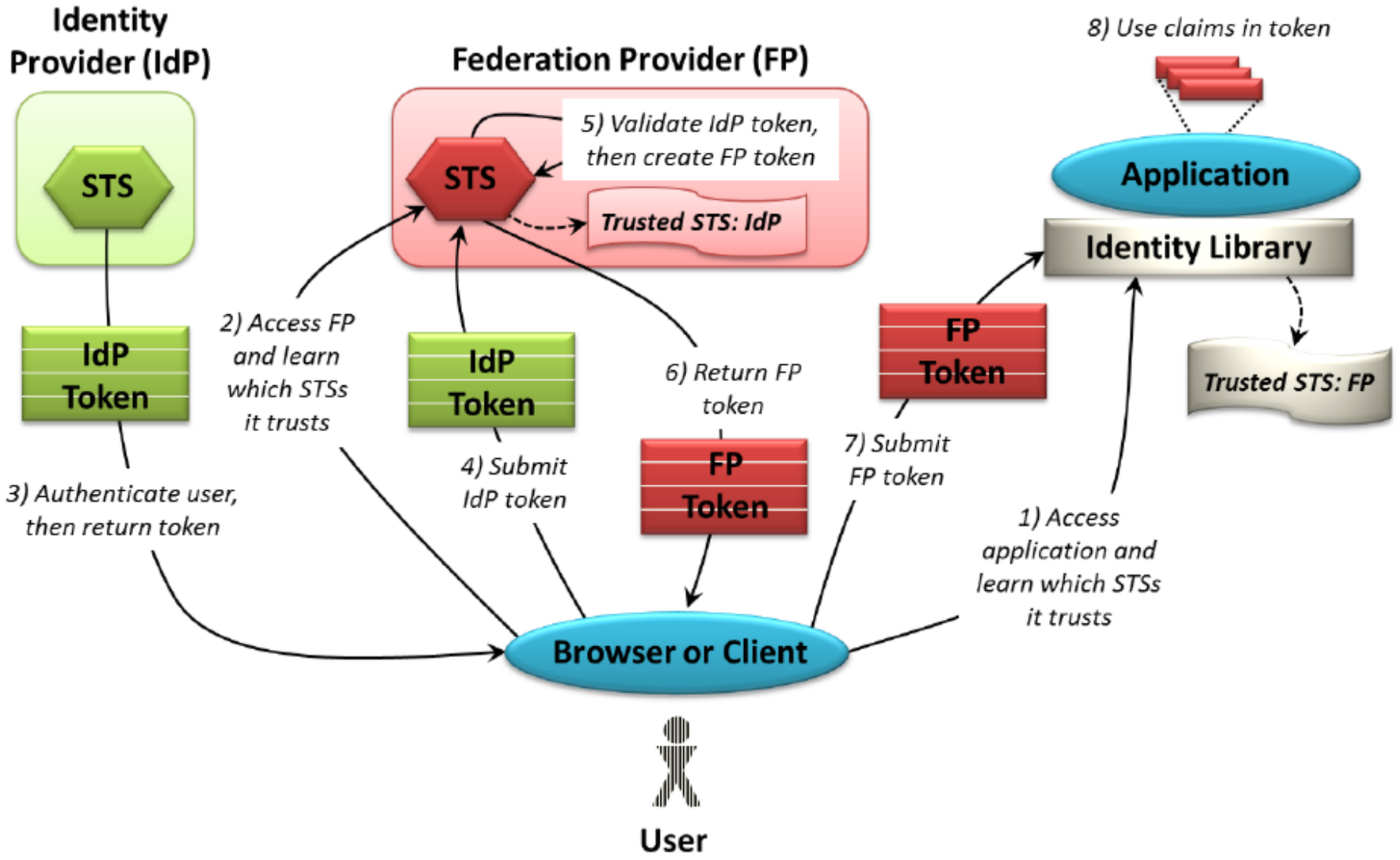
Tokens


Token
Claim 1
Claim 2
Claim 3
. . .
Claim n
Signature
Example Claims
Name
Role
Age

Basic Communication Pattern



Identity Federation



SAML

SAML

- Security Assertion Markup Language
- XML-based
- Supports single sign-on (SSO)
- Requires mutual trust between IdP and SP
- Multiple bindings, not just HTTP
- Supports identity provider-initiated authentication

Underlying Standards

- Extensible Markup Language (XML)
- XML Schema (XSD)
- XML Signature
- XML Encryption
- Hypertext Transfer Protocol (HTTP)
- Simple Object Access Protocol (SOAP)

SAML Versions

- SAML 1.0 - 2002
- SAML 1.1 - 2003
- SAML 2.0 - 2005
  - Incompatible with SAML 1.x
  - Renamed XML namespaces, elements and attributes
  - New bindings
  - New protocols

SAML Assertions

```
<saml:Assertion ID="_3c39bc0fe7b13769cab2f6f45eba801b1245264310738"
   IssueInstant="2009-06-17T18:45:10.738Z" Version="2.0">
   <saml:Issuer Format="urn:oasis:names:tc:SAML:2.0:nameid-format:entity">
      https://www.salesforce.com
   </saml:Issuer>
```

Assertion A was issued at time t by issuer R regarding subject S provided conditions C are valid.

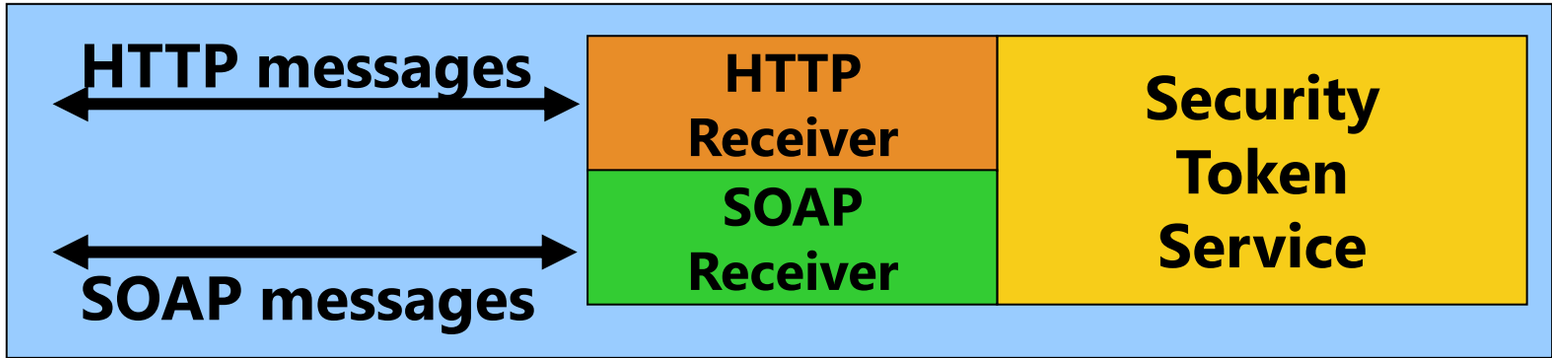
- Authentication statements
- Attribute statements
- Authorization decision statements

SAML Sign-In Protocol

https://sts.cloudready.ms/adfs/ls/?SAMLRequest=jZFRT4Mw-FIX%2FCun7KC3OjWaQ4PbgkqIkoA%2B%2BmAKdNCkt9hZ1%2F14Gmk-wfFI%2Fv%0APfc7p6cr4K3qWNq7Ru%2FFWy%2FAeZ%2Bt0sDGRYx6q5nhIIFp3gp-grmJ5erdj1A9Y%0AZ40zIVHISwGEddLotdHQt8Lmwr7LSjzudzFqnOuAYQyNLP1Kmb62gtd-HvwWc%0AD6PSKOEaH8DgE5ni7CEvkLcZokjNT9AzhIM%2FBF4fACvAyNtuYvRSRSI-iZXIN%0AwrIY0CriSxKGhNLDFeXhYjkfZAC92GpwXLsY0YCEM0JnQVQESxaEj-CyekZd9%0AP%2BxG6Irq18stIJMI2G1RZLMp%2FJOwMAYfBChZnbpko7E9a%2FcyIv9UipJ%2FF-bjC%0AZy6TZcfuB%2Bx2kxkIq6OXKmU%2B1sOpEzEiCCfTye%2FfT74A%0A&RelayState=cookie%3A29002348&SigAlg=http%3A%2F%2Fwww.w3.org%2F2000%2F09%2Fxmldsig%23rsa-sha1&Signature=M0xoWQfcN3Yp94T2HiqIdJzEkxYqGc6hhopqi8xOI%2B2BtPSLufFD-dQIF7z6Xjm6XdLq1MH9Av5xz2QWYs84ZYhIG3fHtZCjjaoI2wZqpIRszHIa%2BjtZoW20NGDe-pDsCRT0AKNkhe%2B4Yj3LshrM6EX5O3obx2Mypy8EcsoURkTF3kf1dwKqsGA3ka7ehbRmUQGJUXD0u4iF-Bog7YgkL4Q9FYMTanZeRo2X4%2FkAeNxT8ormKWJfYnAzg0F4Ku60zDd5N7jYu4XeyOsXDthEFI5H4WYu-cAprREI2hgSUI21J782kKzrsIaIIaJ5BKPIO50NPCIb5Sf6Zw4maLpZrFEfrw%3D%3

WS-Federation

- OASIS Standard from 2006, drafted by IBM, Microsoft, RSA,…
- Uses SAML Tokens
- Based on WS-Security, WS-Trust and other WS-* standards
- Supports automatic metadata discovery and certificate roll-over
- Defines common claims
- Two profiles of the model defined
  - Passive – for web browser clients
  - Active – SOAP clients



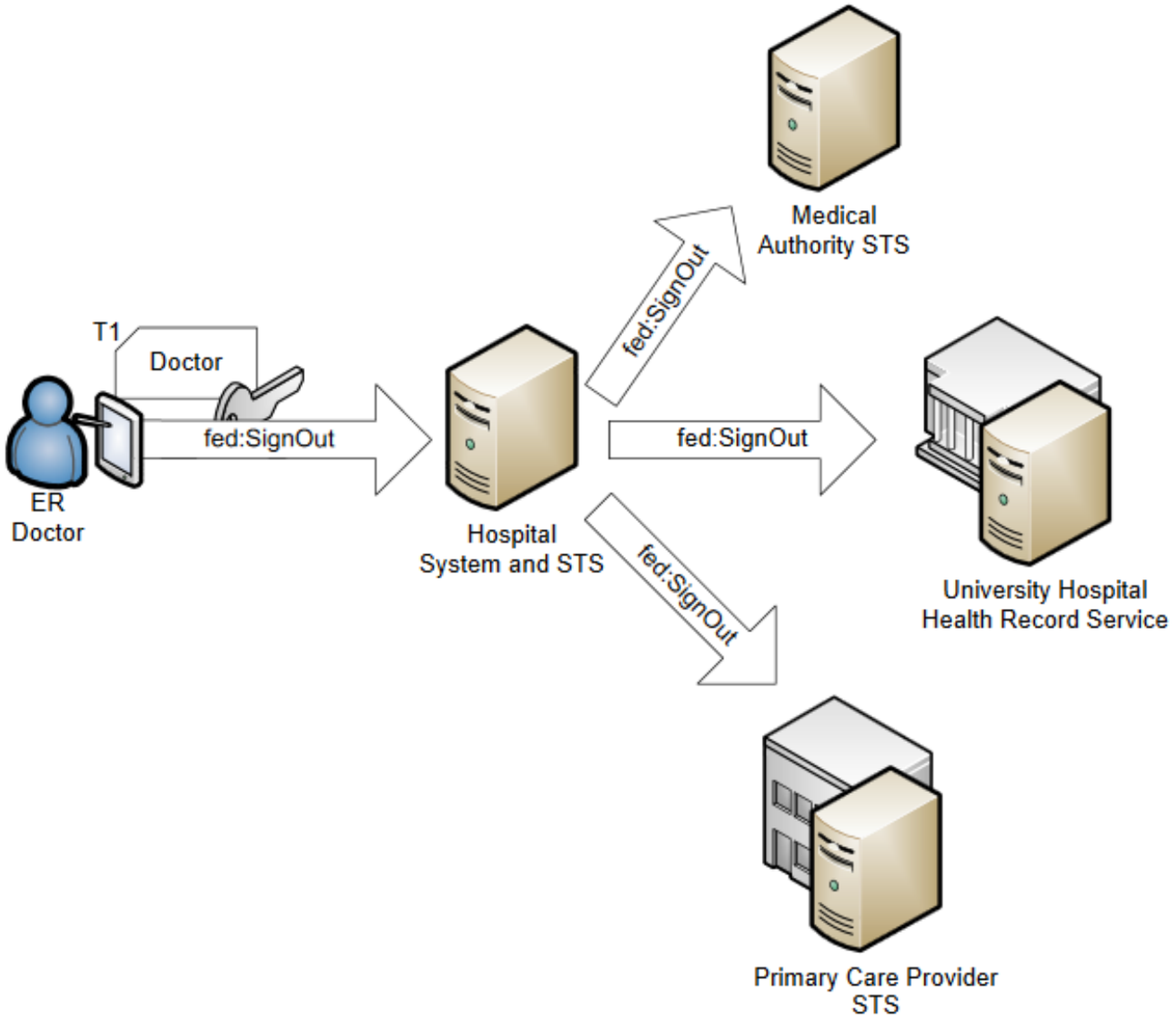
WS-Federation SignIn

https://sts.cloudready.ms/adfs/ls/?wa=wsignin1.0&wtrealm=https%3a%2f%2fclaim-sweb.cloudready.ms&wctx=rm%3d0%26id%3dpas-sive%26ru%3d%252f&wct=2014-10-21T22%3a15%3a42Z

Let's break-out these parameters:

- **Wa=signin1.0:** This tells the ADFS server to invoke a login for the user.
- **Wtrealm:** This tells ADFS what application I was trying to get to. This has to match the identifier of one of the relying party trusts listed in ADFS.
- **Wctx:** This is some session data that the application wants sent back to it after the user authenticates.
- **wct:** This is the exact time I tried to gain access to the application.

Single SignOut



FederationMetadata

- Entity (STS) ID
- Token signing certificates
- WS-Federation endpoint URL
- SAML protocol endpoint URL
- …

JSON Web Token

JSON Web Token

- Defined in IETF RFC7519 from May 2015
- Inspired by SWT
- Based on
  - **JSON Web Signature (JWS, RFC7515)**
  - **JSON Web Encryption (JWE, RFC7516)**
- Very compact
- Can use different encryption schemes

Token Structure

Header: { typ: 'JWT', alg: 'HS256' }
Payload/Claims:
{
    user: john,
    admin: true,
    exp: 8.10.2016, 15:27
}
Signature

=> BASE64

Encoded and Signed Token

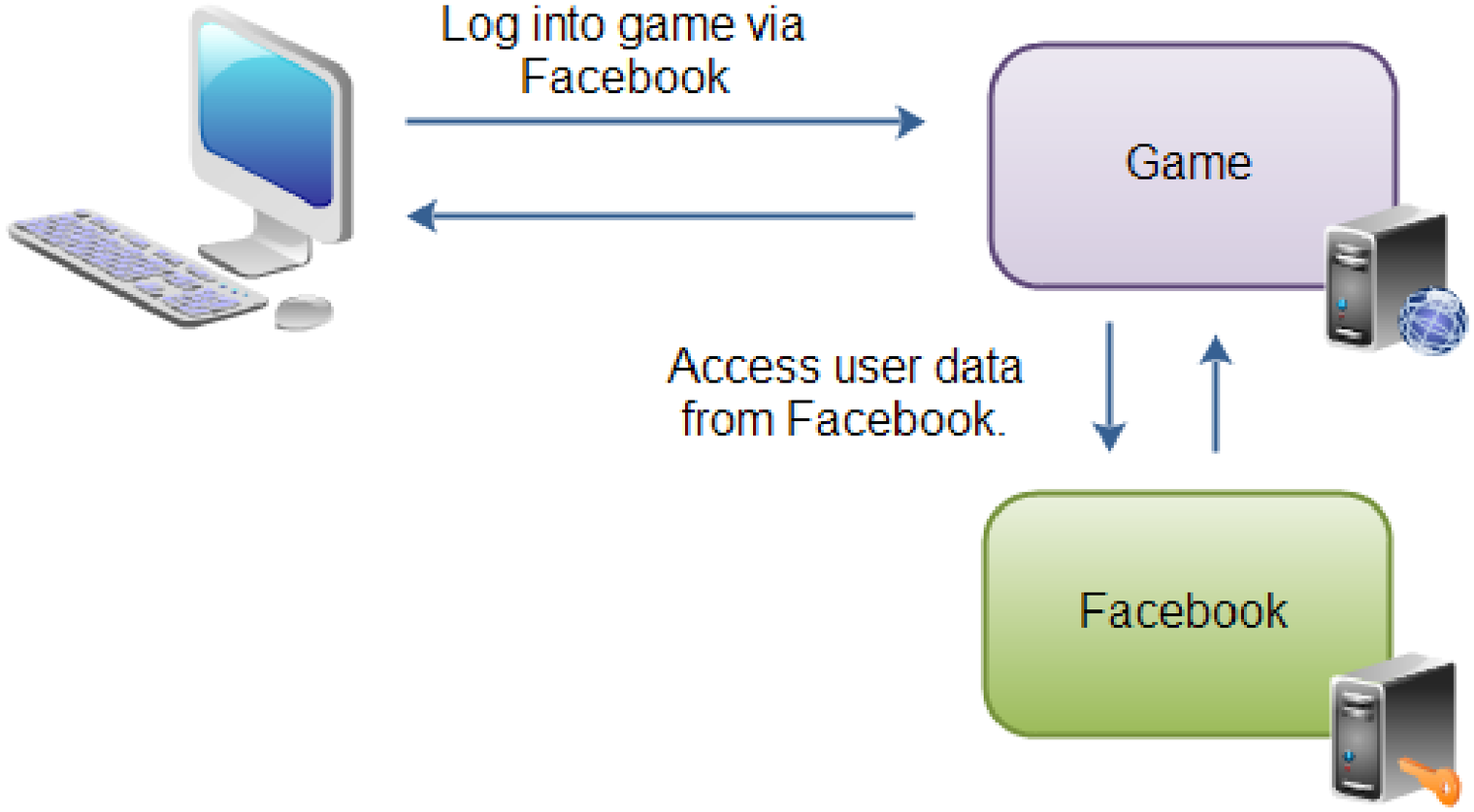
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4
gRG9lIiwiaXNTb2NpYWwiOnRydWV9.
4pcPyMD09olPSyXnrXCjTwXyr4BsezdI1AVTmud2fU4

OAuth

OAuth / OpenID Connect

OAuth

Used to delegate user authorization to a 3rd-party service provider



History

- 2006 – Twitter missing delegation in OpenID
- 2007 – OAuth Core 1.0 (Twitter, Google,…)
- 2010 – OAuth 1.0 (IETF RFC 5849)
- 2012 – OAuth 2.0
  - Framework - RFC 6749
  - Bearer Token Usage - RFC 6750
  - Threat Model and Security Considerations - RFC 6819
  - …

OAuth Token Types

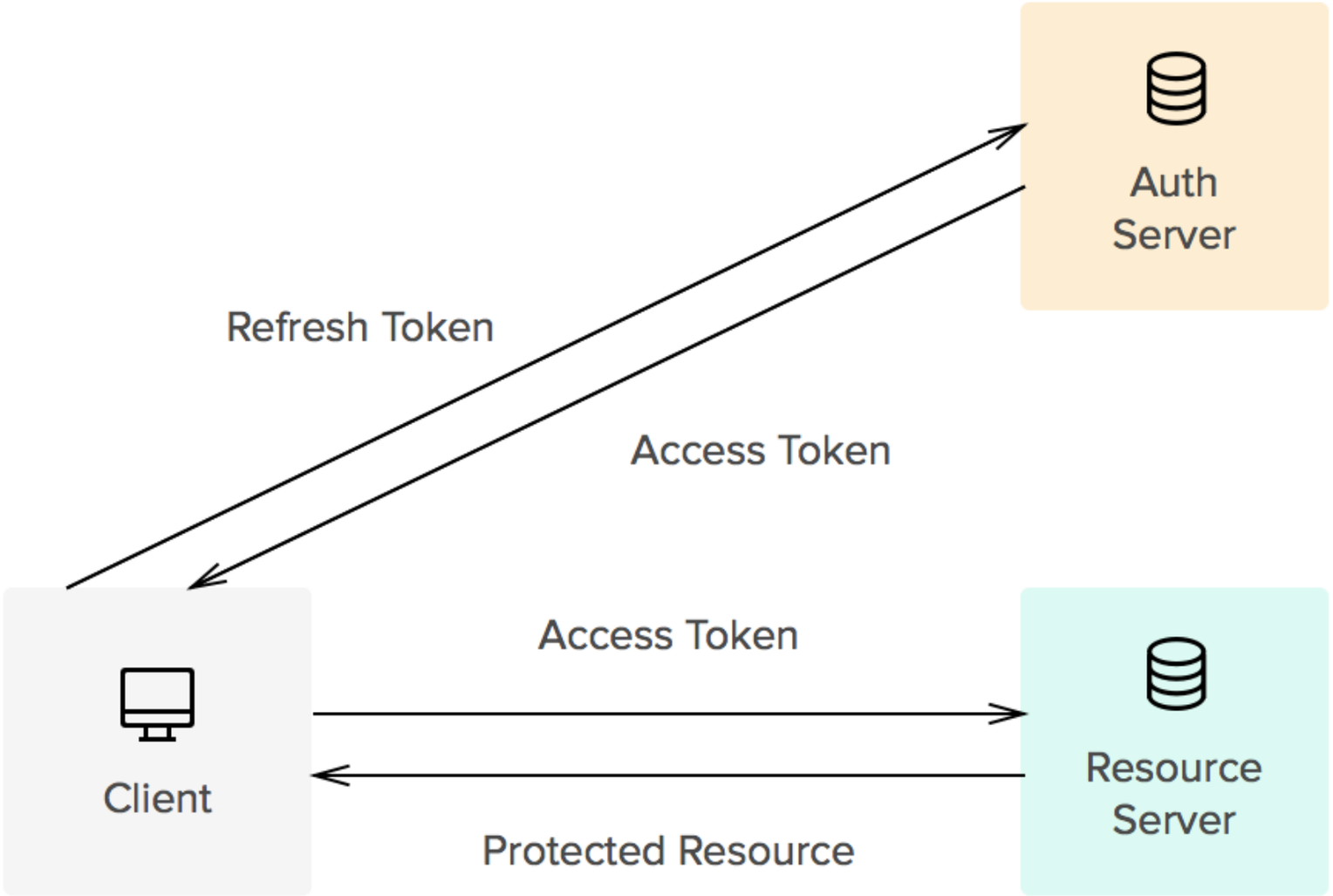


Image Source: https://auth0.com/blog/refresh-tokens-what-are-they-and-when-to-use-them/

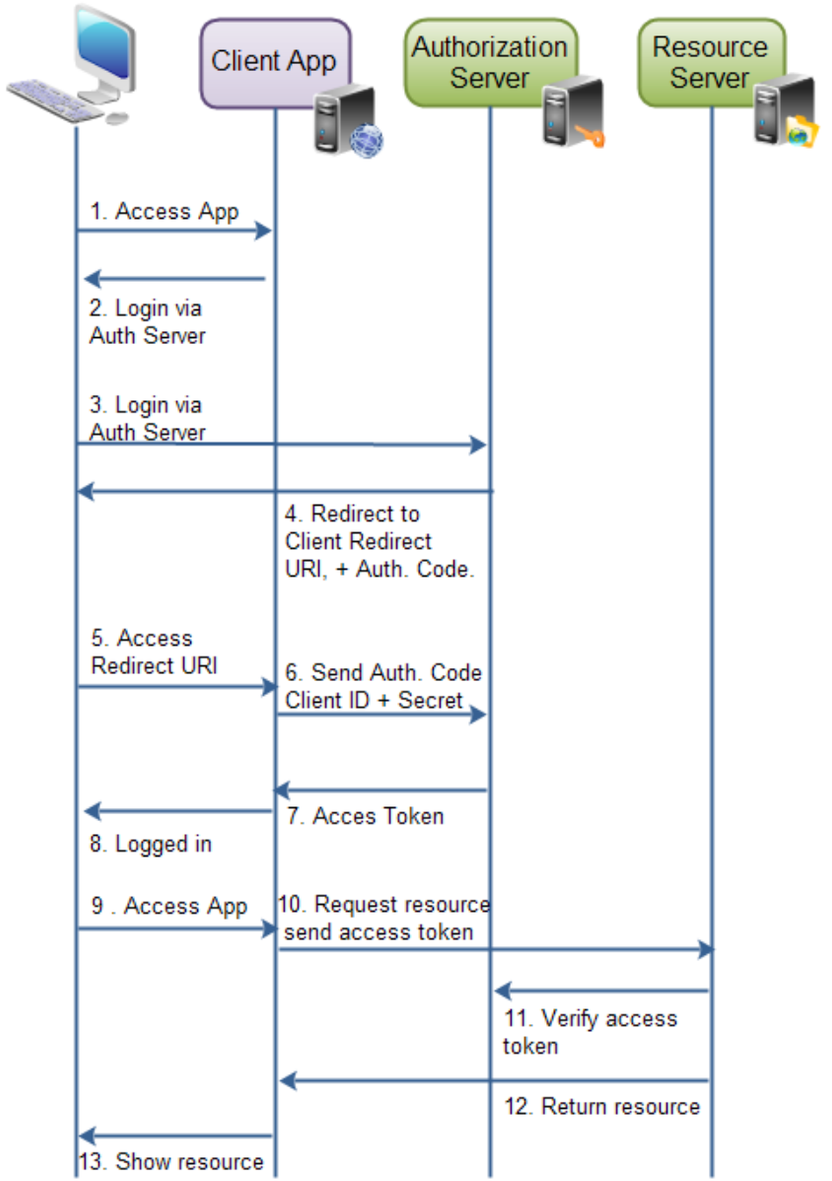
OAuth Scopes (GitHub)

- read:org
- write:org
- read:discussion
- write:discussion
- read:packages
- write:packages
- delete:packages
- …

OAuth 2.0 Grant Types

- Authorization Code
  - Proof Key for Code Exchange (PKCE) Extension
- Client Credentials
- Device Code
- Refresh Token

Authorization Code Flow



Authorization Request

https://sts.cloudready.ms/adfs/oauth2/authorize?response_type=code&client_id=3fb2a37f-4ced-409c-937c-dddd776f4dfd&redirect_uri=https://www.davetestapp.com&resource=https://www.davetestapp.com

Let's break down these parameters:

- **response_type:** tells that ADFS server that I want to perform OAuth and get an authorization code in return.
- **client_id:** The ID of the application I'm trying to get to.
- **Resource:** the URL/URI of the application I'm trying to get to.
- **redirect_uri:** Tells ADFS who to POST the auth code back to

Authorization Prompt

https://authorization-server.com/auth?scope=create+delete&client_id=2

Example Service Signed in as User Name

Sample App

https://example-app.com by ACME Corp

This app would like to:

Create posts

Delete posts

Cancel Allow

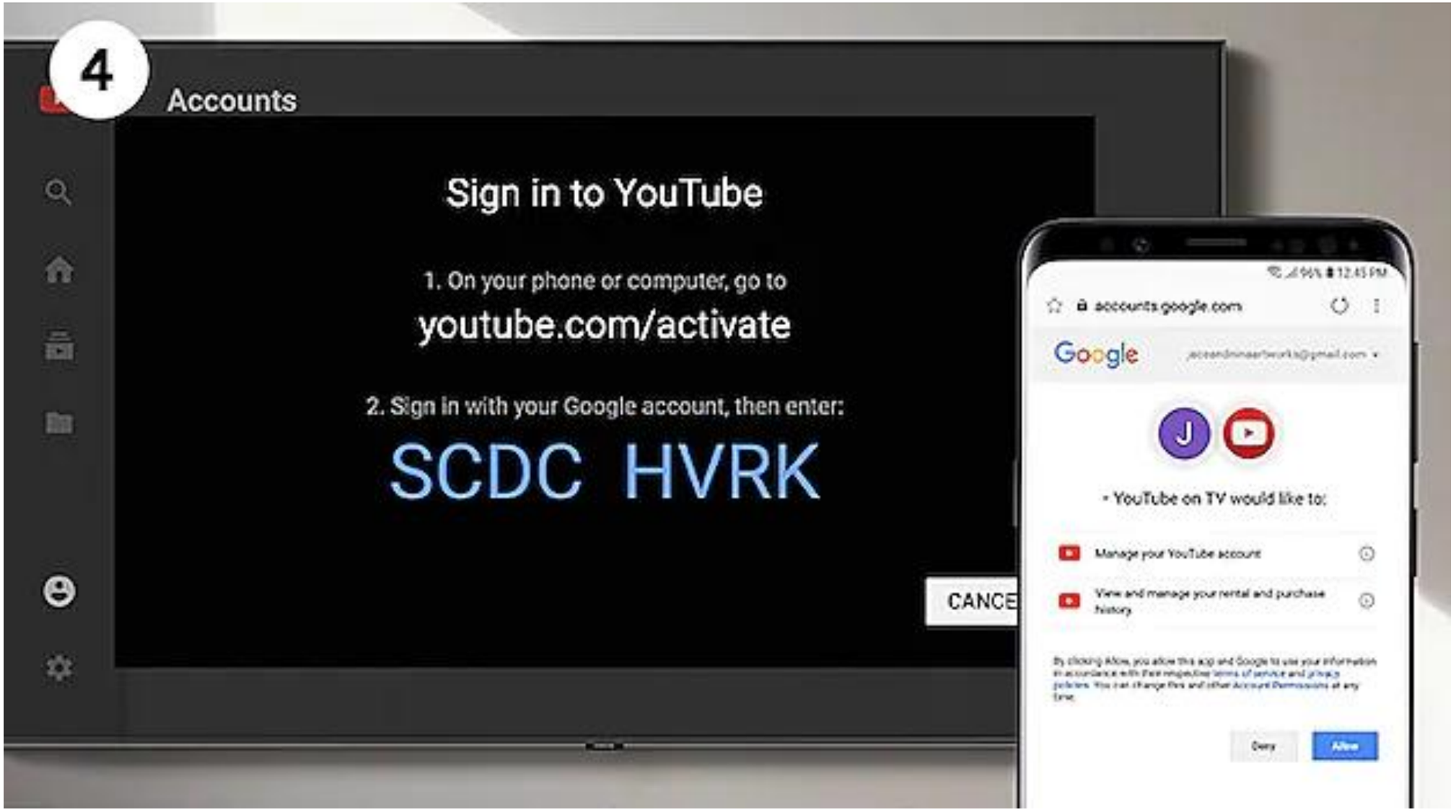
Authorization Response

ms-app://s-1-15-2-1101140336-4090662585-1905587327-262951538-2732256205-1306401843-4235927180?code=AAAAAAAAAAAAAAAAAAAAAAAA.lxt7fs590QgJAJqMtW9C8KN6tLE.SzhnV- R4lbKXj46nwIbFUm6SLnyryJNE72eg3797LwkSfFQsSNpr7E9sUlCYBH52xvLZDAbwwu7qXCMlqCuVciQ0j3P3-l3ep_lOSJOD3LnwDnXb3MPM1UUNcGLxxVeJmeYhEr15BasdkWqGzTYrCJKf4jbdVT0qb4HKEhpD2aQCDwqjeFF8mNwfne_KL1Ve6ZTNwBWS41SauUnbCTM9qzx-MCDWKEPrLmRR14hCxIsaWfrHmiEYbfl4JXyGcJvhUyffcL-UVwJsQSBjHGlbQXlwrb-ejvvZ6me3YC8CLoS2pvXAMzbppBfg8YAJbGzBPNplbkjM10A7OKifLT4yqQ

OAuth 2.0 Client Credentials Grant (Confidential Clients)

```
POST /token HTTP/1.1
Host: authorization-server.com

grant_type=client_credentials
&client_id=xxxxxxxxxx
&client_secret=xxxxxxxxxx
```

OAuth 2.0 Device Authorization Grant



OpenID Connect