

# Designing stack agnostic, modern, secure architectures

Eugene Pilyankevich,  
Chief Technical Officer, **Cossack Labs**

# #whoami / Speaker intro

- Infosec since mid-90s.
- Designed, supervised development of banking data processing, risk management DSS, cryptographic libraries, high-load services.
- Protected some state secrets, banking data, critical infrastructures, patient records, transactions and payment data.
- CTO, co-founder at Cossack Labs - data security solutions provider ([www.cossacklabs.com](http://www.cossacklabs.com))
- Life-long interest in how big systems fail and stand against failure.

# Designing **stack agnostic, modern, secure** architectures

Sounds a bit like CAP theorem, isn't it?

**stack agnostic, modern, secure**



# Sounds a bit like CAP theorem, isn't it?

1. Stack agnostic = Architecture that is not limited with certain implementations or availability of certain types of infrastructure;

# Sounds a bit like CAP theorem, isn't it?

1. **Stack agnostic** = Architecture that is not limited with certain implementations or availability of certain types of infrastructure;
2. **Modern** = Architecture that enables modern design approaches and addresses modern, relevant risks and threat models;

# Sounds a bit like CAP theorem, isn't it?

1. **Stack agnostic** = Architecture that is not limited with certain implementations or availability of certain types of infrastructure;
2. **Modern** = Architecture that enables modern design approaches and addresses modern, relevant risks and threat models;
3. **Secure** = Resilient against chosen risks;

1. Stack agnostic

2. Modern

3. Secure



SA + M + S

How do we get to SA + M + S ?

**Step 1. Understand goals of security architecture, why do we need it, what is the value and the benefit?**

**Step 2. Understand necessary design and implementation steps in practical context.**

**Step 3. Understand and overcome limitations during both design and implementation.**

# How do we get to SA + M + S ?

Part 1. **Why do we need security architectures?** Why can't we just build ISMS or just address OWASP Top 10?

Part 2. **Building blocks of security architecture.** Risk management, attack surface, balancing tradeoffs.

Part 3. **Typical approaches to resolving conflicts** and overcoming limitations while preserving SA, M & S.



COSSACK  
LABS

WHY?

WHY?

WHY?

WHY?

WHY?



# Why we need security architecture?

WHY?

WHY?

WHY?

WHY?

WHY?

Let's start with a story.



# Not an easy target

ISO 27000

A+ rating in banking security compliance

Annual audits and frequent pentests

... in 2008 we pretty much ahead of the game, we thought.

# Perfect user fraud prevention solution.

## Defenders

- Cookie / Session / IP binding
- Concurrent session matching
- Concurrent query analysis
- Rate limiting
- Terms of service enforcement
- Browser fingerprinting
- Complex JS wizardry

## Attackers

Yeah, right, let's see what they came up with now.

# Perfect user fraud prevention solution.

## Defenders

- Cookie / Session / IP binding
- Concurrent session matching
- Concurrent query analysis
- Rate limiting
- Terms of service enforcement
- Browser fingerprinting
- Complex JS wizardry
- **Charge customers per request**
- **Void abuser's contracts**

## Attackers



# Perfect user fraud prevention solution.

## Defenders

- Cookie / Session / IP binding
- Concurrent session matching
- Concurrent query analysis
- Rate limiting
- Terms of service enforcement
- Browser fingerprinting
- Complex JS wizardry
- **Charge customers per request**
- **Void abuser's contracts**

## Attackers

Account misuse and fraud drop below 5% within 180 days.

# Perfect user fraud prevention solution.

## Defenders

- **Cookie / Session / IP binding**
- **Concurrent session matching**
- **Concurrent query analysis**
- **Rate limiting**
- **Terms of service enforcement**
- **Browser fingerprinting**
- **Complex JS wizardry**
- **Charge customers per request**
- **Void abuser's contracts**



**"Prevent it with more code" – engineer's decision.**

# Perfect user fraud prevention solution.

## Defenders

- Cookie / Session / IP binding
- Concurrent session matching
- Concurrent query analysis
- Rate limiting
- Terms of service enforcement
- Browser fingerprinting
- Complex JS wizardry
- **Charge customers per request**
- **Void abuser's contracts**



“Prevent it with more code” – engineer’s decision.



“Prevent it closer to the risks” – manager’s decision

# Now prevent injections on public front

## Defenders

- Input sanitization: front-end
- Input sanitization: back-end
- mod.security config with 2K LOC of custom rules.

## Attackers

The front-end is written in PHP, yeah right.

# Now prevent injections on public front

## Defenders

- Input sanitization: front-end
- Input sanitization: back-end
- mod.security config with 2K LOC of custom rules.
- **Prepared statements.**
- **Materialized views.**
- **Domain model, 4-layer validation.**

## Attackers

Why it stopped failing in new funny ways now?



# Now prevent injections on public front

## Defenders

- Input sanitization: front-end
- Input sanitization: back-end
- mod.security config with 2K LOC of custom rules.
- **Prepared statements.**
- **Materialized views.**
- **Domain model, 4-layer validation.**



Security engineer's decision.

# Now prevent injections on public front

## Defenders

- Input sanitization: front-end
  - Input sanitization: back-end
  - mod.security config with 2K LOC of custom rules.
  - **Prepared statements.**
  - **Materialized views.**
  - **Domain model, 4-layer validation.**
- 👉 Security engineer's decisions.
- 👉 System architect's decisions.

# Why do large companies struggle with this?

- Equifax.
- Heartland Payment Systems.
- JP Morgan.
- RSA Security.
- **Operation Aurora victims:**  
Google, Juniper, other non-confirmed high-profile targets.

# Why do large companies struggle with this?

- “Big companies are hard, big infrastructures are harder to enforce good policies in”
- “Unexpected attack vector under novel threat model accompanied with forces we were not yet prepared to meet”

# Why do large companies struggle with this?

- “Big companies are hard, big infrastructures are harder to enforce good policies in”
- “Unexpected attack vector under novel threat model accompanied with forces we were not yet prepared to meet”
- **On a long enough timeline, the survival rate for everyone drops to zero**

# Why do large companies struggle with this?

- “Big companies are hard, big infrastructures are harder to enforce good policies in”
- “Unexpected attack vector under novel threat model accompanied with forces we were not yet prepared to meet”
- On a long enough timeline, the survival rate for everyone drops to zero
- ͇(ツ)͇

# **WHY?**

Humans are unpredictable

Technology is broken

Poor design decisions



Humans are unpredictable

Technology is broken

**Poor design decisions**

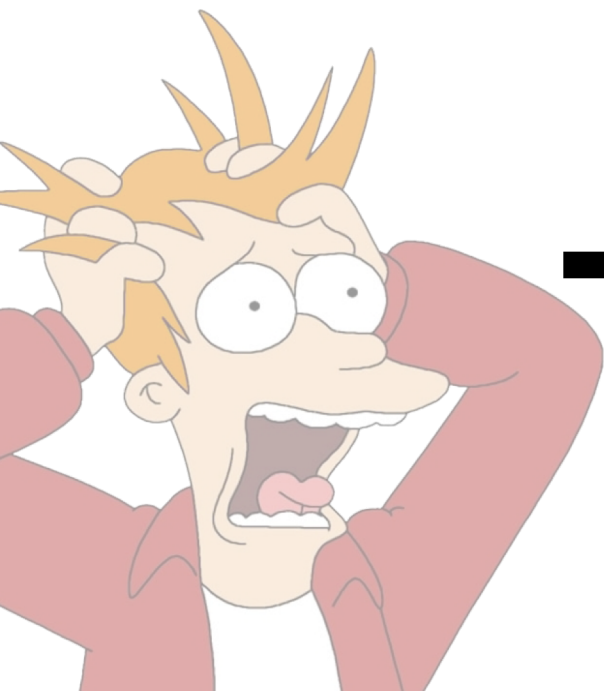
# WHY?



# Poor design decisions

“How to get this security goal done  
and that security concern eliminated?”

# WHY?



# Security...

Has negative business value\*

Is hard to grok\*

Is confusing and contradictory\*

# Security...

Has negative business value\*

Is hard to grok\*

Is confusing and contradictory\*

**Unless you're employed in the infosec industry, where it gets even worse.**

**WHY?**

---



You never know if something is secure or not



You never know if something is secure or not  
... until it's broken.

You never know if something is secure or not  
... until it's broken.

**Then it's definitely not secure.**

# 4 types of knowing

Known Known

Known Unknown

Unknown Known

Unknown Unknown

# 4 types of knowing

Known Known

Known Unknown

Unknown Known

Unknown Unknown

# 4 types of knowing in security

Confusion

Doubt

Fear

Risk aversion

WHY?

---



Thinking about 100 things at the same time is quite frustrating.

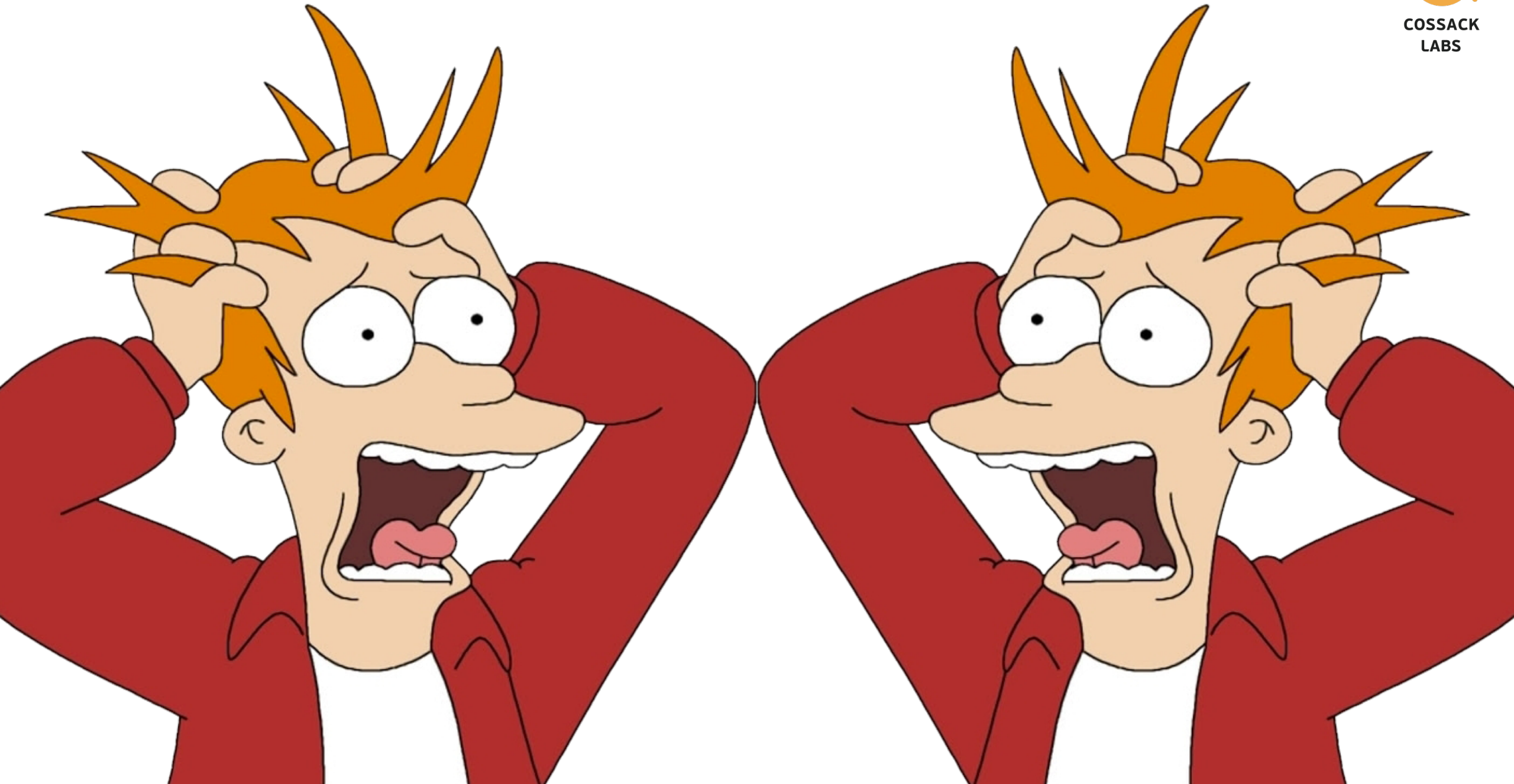
In absence of clear mental model people make poor decisions about risky and complex systems because risk brings affect & bias.

In absence of well-communicated design principles and acceptance criteria mind is prone to emotional affect.

Ability to think systems and ability to think risk is quite domain-specific if you're not conscious about it.

**People make more mistakes  
about risky things under  
pressure in absence of simple  
guiding principle.**





# Remember story I started with?

Manager's decisions.

Security engineer's decisions.

Software engineer's decisions.

System architect's decisions.



# Remember story I started with?

Manager's decisions.

What is bad for us?

Security engineer's decisions.

How to prevent that "bad"?

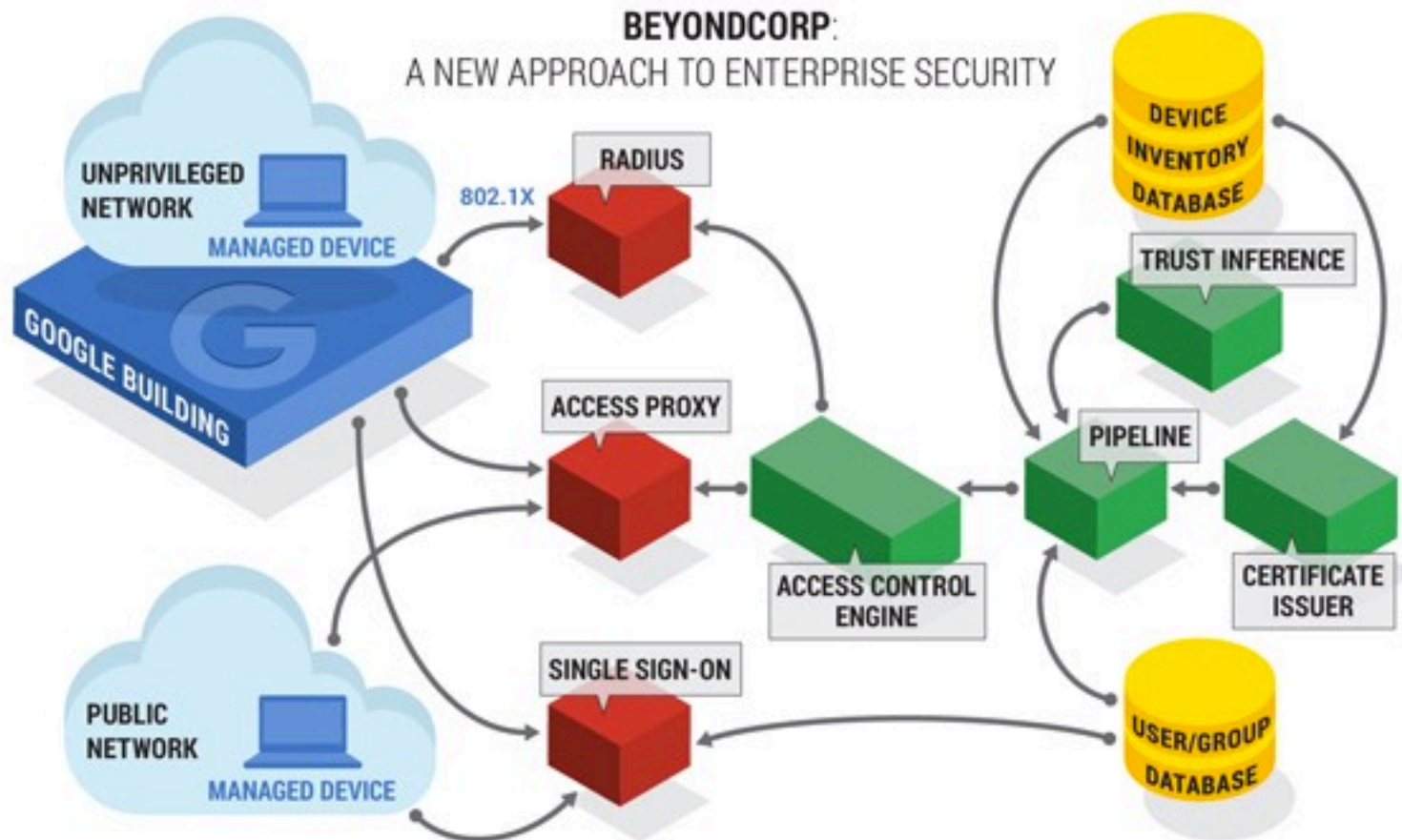
Software engineer's decisions.

What my stack suggest to do?

System architect's decisions.

What is the right systematic way?

# Remember the giants?



BeyondCorp components and access flow

Google: revised the AC architecture.

<https://cloud.google.com/beyondcorp/#researchPapers>



# Security architecture 101:



## Intro

# Goals of security architecture?

We want **understandable and implementable decision system** that allows us to:

1. Prevent damage to business
2. Manage risks cost-efficiently

# What is security architecture?

Combination of security decisions.

# What is security architecture?

Combination of security decisions, which makes actual system's risks manageable.



# What is security architecture?

Combination of security decisions, which makes actual system's risks manageable in a chosen manner, efficiently.

# What is security architecture?

Combination of security decisions, which makes actual system's risks manageable in a chosen manner, efficiently, while maintaining all other quality attributes of a system on acceptable level.

# How to design the security architecture?

- Understand and manage the risks
- Understand and manage attack surface
- Balance tradeoffs



**Before we do these three things,  
security effort is just re-painting  
this door in fancy colors.**



# Security architecture 101:

Intro

👉 Understanding risks

Building secure architecture is similar to building scalable and resilient architecture.

It's the set of risks that is different, but the approach is the same – you **design against the chosen valid risks** for you.

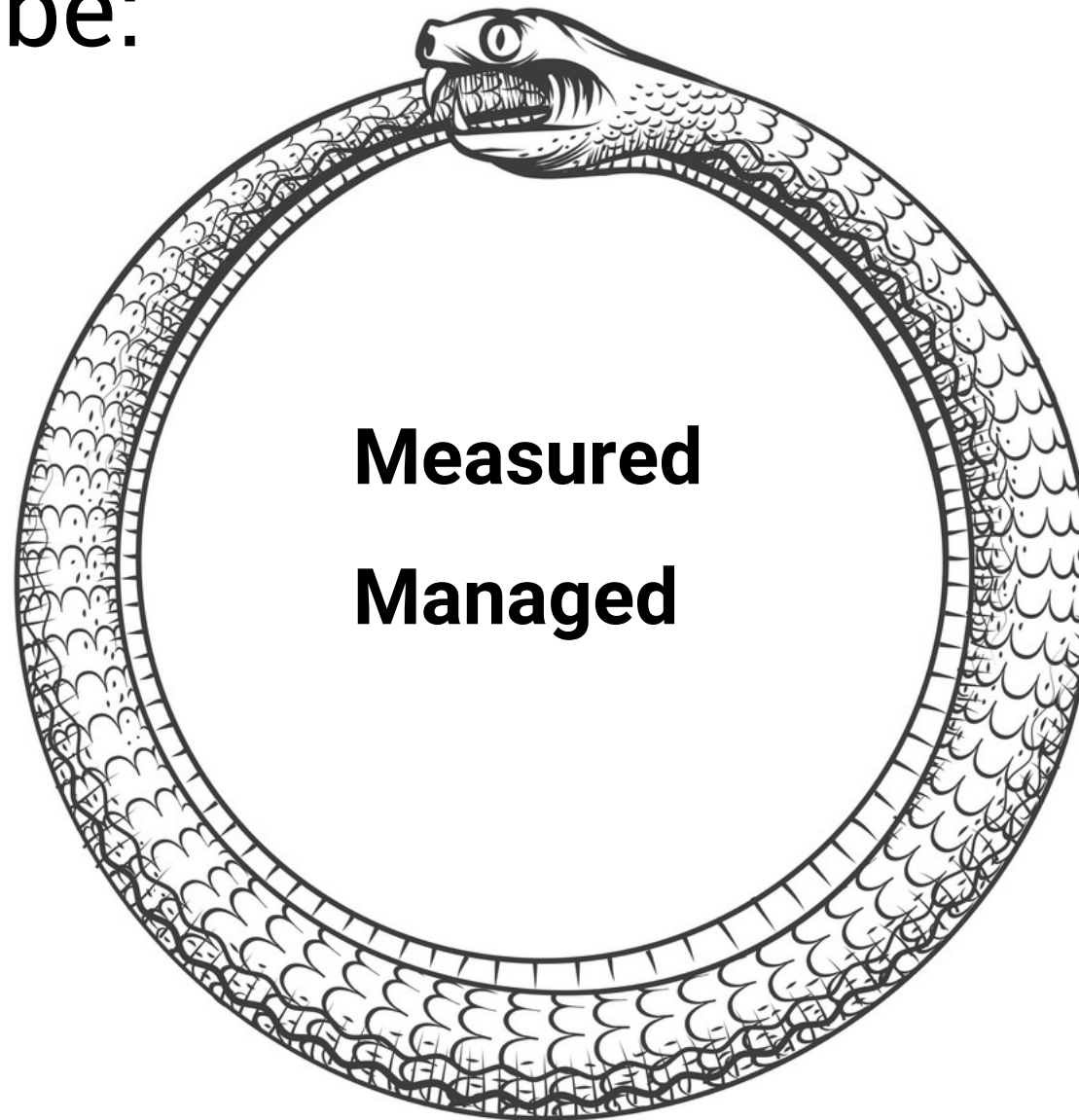
You?



NASA

US Navy

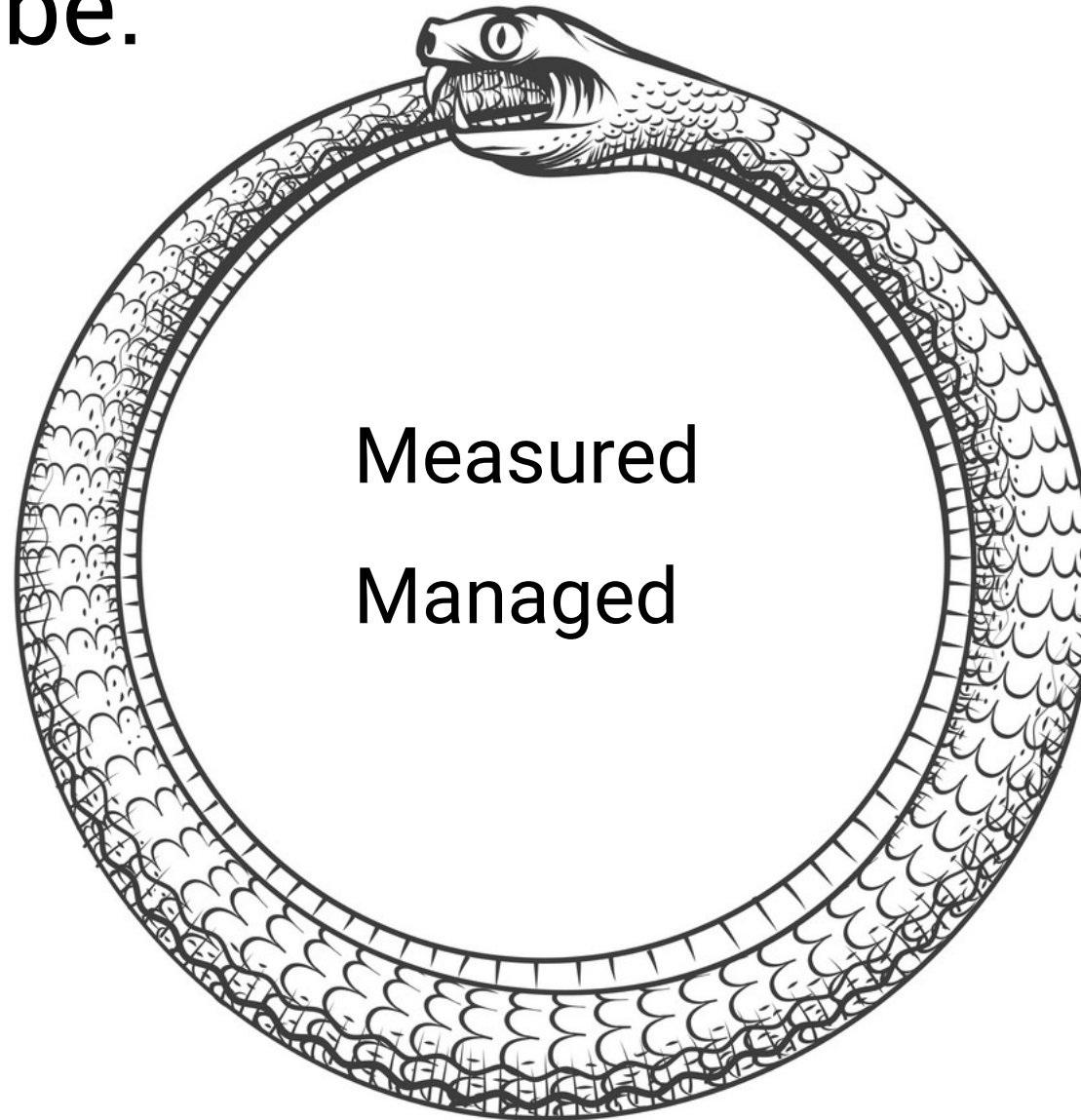
Risk should be:





Risk should be:

**Quantitatively**



**Adequately**

**Appetite/governance**

**Identification**

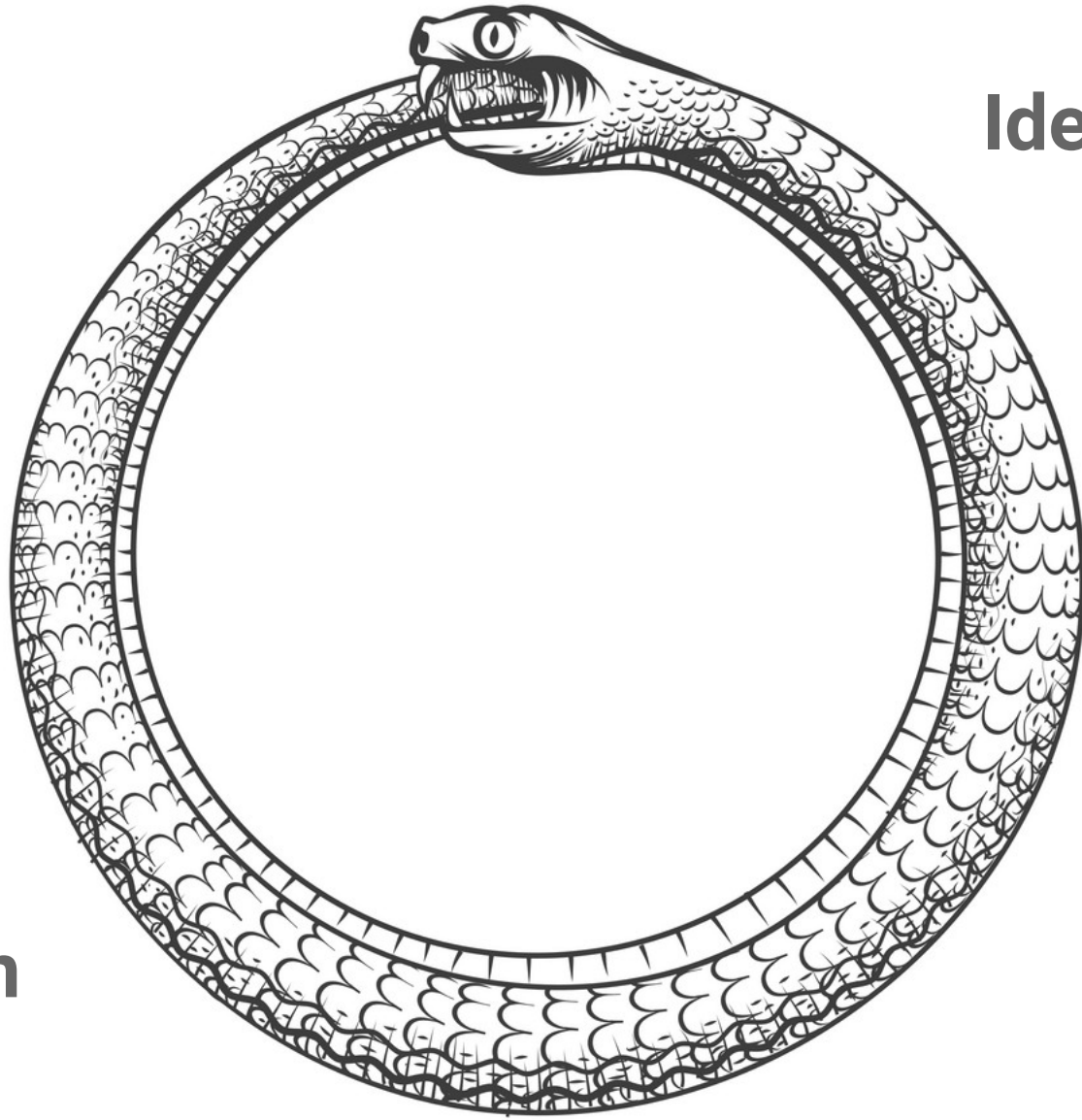
**Assessment**

**Treatment**

**Monitoring**

**Acceptance**

**Mitigation**



# Risk management

## Questions:

- What is more important to protect and how? Why?
- Should we spend more on this or on that?

## Valuable approaches:

- OWASP RAF
- FAIR
- NIST RMF
- COBIT 5
- OCTAVA

Risks ~ Problem probability  
Probable damage

Remember:  
One in a million is next Tuesday.



# Security architecture 101:

Intro

Understanding risks



Understanding attack surface

# Understanding attack surface



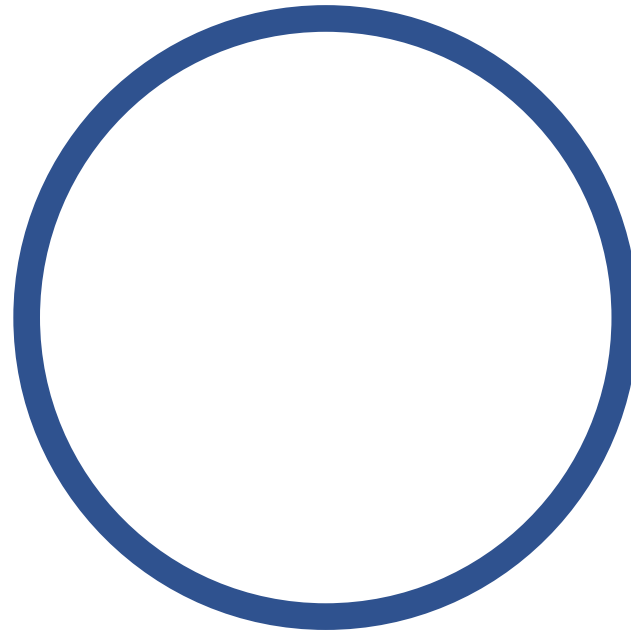
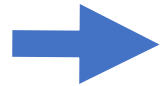
Bad people



Bad people

# Understanding attack surface

Attack  
surface





# Understanding attack surface

Attack Surface is every possible way attacker can induce chosen type of loss to your system.

# Attack surface is your friend

Instead of **“protecting every system”**,  
you can to **focus on protecting the attack  
surface.**

# Understanding attack surface

- Attackers look for assets.
- Defenders protect boxes.

# Understanding attack surface

- Attackers look for assets.
- Defenders protect boxes.
- Attackers think in graphs.
- Defenders think in lists.

# Understanding attack surface

**Prioritized by damage ☹️**

- Attackers look for assets.
- Attackers think in graphs.
- Defenders protect boxes.
- Defenders think in lists.

**Not prioritized by risk ☹️**

## Note: An unfair asymmetry

- **To win against attacker**, you need to ensure that every vector on attack surface is protected.
- **Attacker to win against you**, needs to find one (in worst case several) unprotected attack vectors.

# Managing attack surface

- **Assessing** attack surface.
- **Minimizing** attack surface.
- **Controlling** attack surface.
- **Monitoring** attack surface.
- **Drills.**



# Security architecture 101:

Intro

Understanding risks

Attack surface

 **Balancing tradeoffs**





# Balancing tradeoffs

Risk impact

Cost

# Balancing tradeoffs

Risk impact

Cost  
Usability

# Balancing tradeoffs

Risk impact

Cost  
Usability  
Maintainability

# Balancing tradeoffs

**Risk impact**

Cost  
Usability  
Maintainability  
**Flexibility**

# Balancing tradeoffs

- This is not A vs B relationship: security + usability.

# Balancing tradeoffs

- This is not A vs B relationship: security + usability.
- **Pick your battles** – you can't have all NFRs in a perfect shape.

# Balancing tradeoffs

- This is not A vs B relationship: security + usability.
- **Pick your battles** – you can't have all NFRs in a perfect shape.
- Seek solutions that have:

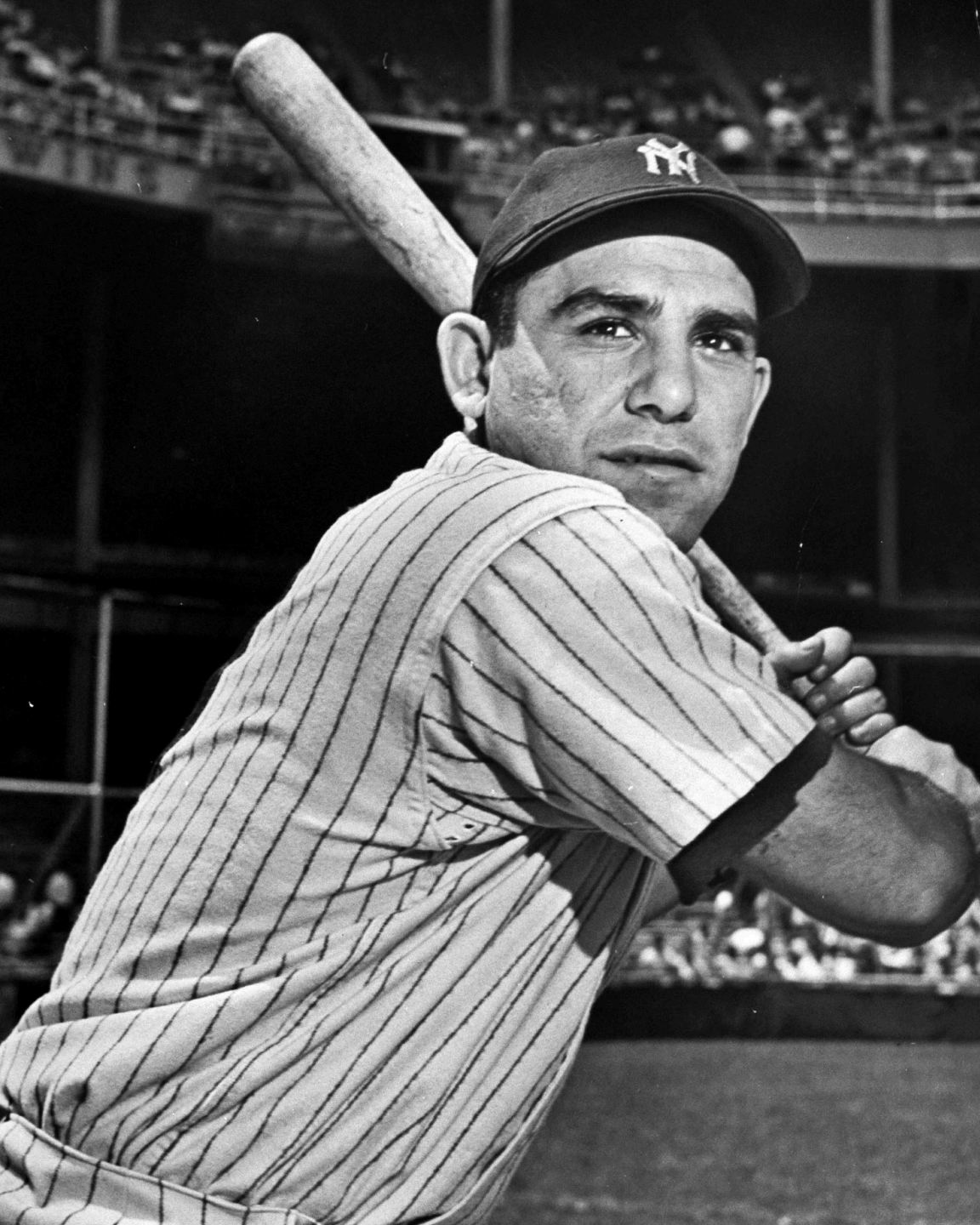
Both **acceptable risk impact** and **acceptable baseline qualities** for all NFRs.





# Designing for security: understanding and overcoming limitations





**In theory, there is no  
difference between  
theory and practice.**

**In practice, there is.**

**Yogi Berra,  
New York Yankees,  
catcher, coach and manager**

Attack surface is always too big.

# Attack surface is always too big.

- Real attack surface is always just crazy big.
- Variety of technologies, tools and assets is crazy big.
- **The only thing that is not crazy big?**
- **Staff and security budget.**

# Attack surface is always too big.

- **Example:** two power grid monitoring efforts.

Humongous limitations, mad scale, bad legacy.  
... security?

# Attack surface is always too big.

- **Example:** optimizing SIEM coverage.

Need more signals, got less eyes.

Review risk model and decrease the scope (for real).

**Prioritize!**

You can't fix everything.

**Prioritize!**

You can't fix everything.  
Choose your battles.



# Is it secure?

## Trust levels

1. Ultimate “secure”.
2. Nothing is “provably secure” in absolute terms.
3. Raising the bar, raising cost
4. Controlling attack flow.

Sometimes requirements  
**conflict with each other!**

**Conflicts arise** when each problem / risk has separate solution / control.

**Conflicts disappear** when solutions in system address root causes of problems and risks.

<https://ivychapel.ink/posts/on-avoiding-band-aid-security/>

- **Example:** optimizing SIEM coverage.  
Data leakage through audit logs.

- **Example:** optimizing SIEM coverage.

Data leakage through audit logs.

PCI logging requirements vs GDPR requirements.

- **Example:** optimizing SIEM coverage.

Data leakage through audit logs.

PCI logging requirements vs GDPR requirements.

Logs are data as well.

- **Example:** optimizing SIEM coverage.

Data leakage through audit logs.

PCI logging requirements vs GDPR requirements.

Logs are data as well.

Should we protect them?

No requirements = infinite rabbit hole.





Things you don't need (yet) to succeed

You don't need most of security tools (yet).

You don't need most of security tools (yet).  
That's just more attack surface.

You don't need most of security tools (yet).

That's just more attack surface.

And more complexity.

### Network Security

**SDN**  
BlackRidge TECHNOLOGY, CERTES, Cybera, Cynet, Cyxtera, NanoSec, SKYPORT SYSTEMS, TEMPERED, VERSA, zentera, zscaler

**DDoS Protection**  
Akamai, Check Point, Cloudflare, Fortinet, Imperva, Neustar, NEXUS GUARD, NSFOCUS, ORACLE, SECURE 64, StackPath

**DNS Security**  
BLUECAT, CISCO, CYREN, efficient IP, Infoblox, neustar, SECURE 64, Threat STOP, VERISIGN

**Network Firewall**  
CLAVISTER, endian, FIREM, FORCEPOINT, FORTINET, Hillstone, SANGFOR, securocloud, SONICWALL, SOPHOS, STORMSHIELD, tufin, untangle

**Deception**  
Attivo NETWORKS, Counter Craft, CyberTrap, Cymermetria, illusive, PACKET VIPER, SMOKESCREEN, TRAPX SECURITY

### Endpoint Security

APERIO, BAYSHORE, BELDEN, CLAROTY CYBERBIT, DRAGOS, endian, FIRMITAS, FORESCOUT, HALO ANALYTICS, Indegy, dimension solutions, NextNine, NOZOMI, PAS, PEP, radiflow, Rhebo, RunSafe, #SCADAfence, sentryo

**Network Analysis & Forensics**  
AWAKE, BRICATA, CGS, CISCO, CloudShark, corelight, CORE SECURITY, Corvil, DARKTRACE, ExtraHop, FIDELIS, GREY CORTEX, IronNet, LUMETA, MixMode, NETSCOUT, PERCH, Plixer, S88, utimaco, VECTRA, VERINT

### Cloud Security

copy, CEQUENCE, Check Point, cisco, ContentKeeper, CYREN, DEFIANT, digicert, distil networks, ERICOM, FORTINET, GOSECURE, FORCEPOINT, GWAVA, iboss, Light Point Security, McAfee, Menlo Security, NAMO-GOO, perimeterx, proofpoint, randed, Reblaze, SH-PE, SHIELD SQUARE, smoothwall, SOPHOS, SPAMINA, Symantec, TREND MICRO, Trustwave, unboxify, whiteaps, zscaler

### Endpoint Detection & Response

Bitdefender, Carbon Black, COMODO, CYBERARK, CYBERBIT, CYBERESON, CYBONET, CYLANCE, deepinstinct, ENDGAME, ENSILO, ERICOM, ESET, F-Secure, FARONICS, FORTINET, HYSOLATE, intego, ivanti, KASPERSKY, McAfee, Microsoft, MORPHISSEC, NYOTRON, OPSWAT, paloalto, panda, SentinelOne, SOPHOS, sparkcognition, STORMSHIELD, Symantec, TEHTRIS, TREND MICRO, vmware, WEBROOT, ZIADANCE

### Application Security Testing

acunetix, beyond, bugcrowd, BUGFINDERS, Fasoo, hackerone, IBM, MICRO FOCUS, PARASOFT, PERFORCE, PORTSWIGGER, Qualys, SecurityCompass, snyk, sonarsource, Synack, Trustwave, VERACODE, Vicarius, wallarm

### MSSP

Advanced MSS & MDR  
CenturyLink, FORESITE, MEGA PATH, NetworkSOLUTIONARY, arizon, RAPID7, Raytheon, Red Canary, RELIAQUEST, UNISYS

### Encryption

ANJUNA, baffie, bascrypter, CipherCloud, CIPHERMARTIN, COVERTIX, CryptoMove, CYBERX, YPHRE, DATALOCKER, ENVEIL, Fortanix, KeyNexus, HENTIS, BnCrypt Cloud, NuCypher, PKWARE, SecurityFirst, STORMSHIELD, THALES, TREND MICRO, virtru, WINMAGICS

### DLP

clearswift, CODE42, COSOVS, Digital Guardian, FIDELIS, FORCEPOINT, INFOWATCH, McAfee, SEARCHINFORM, SOMANSA, Symantec, ZECURION

### Data Privacy

Actifile, BigID, COVATA, D.DAY LABS, D-ID, INTEGRIS SOFTWARE, minereye, nuix, OneTrust, PRIFENDER, SecuPI, SPIRION, TITUS, Uplink, TrustArc, trustohub, VERY GOOD SECURITY, wintrowheel

### Data Centric Security

BlueTalon, CODE42, Datex, dataphy, CyberTech, druva, egress, globalvelocity, IONIC, opentext, PRIVATAR, SECLORE, SPIRION, StorageCraft, THINARX, VARONIS, VERA

### Mobile Security

appdome, BETTER, BlackBerry, BlueCedar, COMMUNITAKE, CyberodAPT, deepinstinct, eMune, Fyde, helixOS, Lookout, mobileiron, NowSecure, NJOBS, pradeo, silent circle, SOTI, Symantec, TeleSign, TRUSTLOOK, VAULTO, vmware, wandera, wickr

### Risk & Compliance

Risk Assessment & Visibility  
Avirin, Coalition, CyberCube, cyberGRX, OESSERVER, KENNA, NEMEMIAH SECURITY, Outpost24, panaseer, PREVALENT, REDSEAL, SKYBOX, tenable, UpGuard, VENAFI, zeguro

Security Ratings  
BITSIGHT, COIRAX, FICO, GUIDEWIRE, WormShield, Panorays, PREVALENT, RiskLens, riskrecon, SecurityScorecard

Security Awareness & Training  
Baracuda, CORNEX, CyberVista, IRONSCALES, KnowBe4, PHISHLABS, proofpoint, SANS

### Security Operations & Incident Response

SIEM  
AT&T Cybersecurity, BLACKSTRATUS, CORRELOG, CYGILANT, DEVO, DNE, EventTracker, exabeam, FORTINET, HanSight, Huntsman, IBM, IGL00SECURITY, JASK, logentrics, LOGPOINT, LogRhythm, logz.io, McAfee, MICRO FOCUS, Palantir, RSA, SAWMILL, SECURONIX, solarwinds, splunk, sumologic, TIBCO, Trustwave

Security Incident Response  
arctos networks, atarlabs, ayehu, CYBERBIT, CYBERRESPONSE, CYBER TRIAGE, D3 SECURITY, DARK LIGHT, DEMISTO, DFLABS, ENCODE, FIREEYE, Microsoft, paloalto, radar, RAPID7, Raytheon, Resilient, SEC3, servicenow, SIEMPRISM, SIFT SECURITY, splunk, SWIMLANE, SYNCRITY

### Threat Intelligence

4iQ, ANOMALI, Blueliv, BlueVoyant, CENTRALIZED NETWORKS, Cisco, Cyberint, digital shadows, DOMAINTOOLS, EclecticIQ, FORTISIGHT, FLASHPOINT, GROUPIBI, HanSight, HYAS, INTEL471, INTSIGHTS, KELA, LOOKINGGLASS, Malware Patrol, NUCLEON, Recorded Future, RiskBased SECURITY, RISKIQ, SenseCy, Sixgill, SpyCloud, SURFWATCH, THREATCONNECT, ThreatMetrix, THREATQUOTIENT, Threat STOP, TRU\*STAR, WEBROOT

### IoT

IoT Devices  
ARMIS, Bastille, CENTRI, Convexum, CyberArk, CREATIS, MDX, CYLLIS, delifor, HONEYWELL, ICON LABS, IMUBIT, KEYFACTOR, LEVEL, MagicCube, MEGAGATE, MOCANA, Regulus, riscure, Rubicon, SECURITHINGS, SEPIO, SENRIO, ZingBox

Automotive  
BlackBerry, Blue, C2A security, CARSDOME, Continental, CYCRO, CYMOTIVE, ENIGMATOS, foretellix, Guard KNOX, HARMAN, Karamba Security, NNG, otonomo, SAFERIDE, Trillium, Upstream

Connected Home  
Bitdefender, CUJO, F-Secure, RUBICA, S.A.M. SECURE NETWORK, Symantec

### Messaging

AGARI, AREA, Barracuda, CYBONET, FORCEPOINT, GreatHorn, IRONSCALES, Microsoft, min, proofpoint, SOPHOS, TREND MICRO, VALINGAIL, wickr

### Identity & Access Management

Authentication  
Averon, BehaviorSec, BIOCATCH, Calsign, Centrify, CLEF, EXOSTAR, FORGEROCK, FUDQ SECURITY, Google, HPR, IDEE, JUMIO, nok, nok, pindrop, plainID, SAASPASS, SaferPass, SECURE PUSH, ShoCard, SILVERFORT, tascent, ThreatMetrix

### Digital Risk Management

OCE CRISP, CYBERSPRINT, digital shadows, DigitalGlobe, EXPANSE, LOOKINGGLASS, NAMO-GOO, PHISHLABS, RISKIQ, SafeGuard Cyber, source, ZEROFOX

### Security Consulting & Services

accenture, ADAPTURE, A-LIGN, appsec, BISHOPFOX, Booz | Allen | Hamilton, BT, CORVID, Deloitte, fishback, GreyCastle, GuidedPoint, IBM, IOActive, Komodo CONSULTING, leidos, MindPoint Group, nccgroup, OPTIV, PwC, REVEALRISK, SECURITYCOMPASS, SERA, STROZ FRIEDBERG

### Security Consulting & Services

accenture, ADAPTURE, A-LIGN, appsec, BISHOPFOX, Booz | Allen | Hamilton, BT, CORVID, Deloitte, fishback, GreyCastle, GuidedPoint, IBM, IOActive, Komodo CONSULTING, leidos, MindPoint Group, nccgroup, OPTIV, PwC, REVEALRISK, SECURITYCOMPASS, SERA, STROZ FRIEDBERG

Good architecture is both decision framework and design guide. It not only addresses the risks, it reduces complexity.

If you're focused on the risks and attack surface of sensitive assets, technology and stack is rarely an issue.

**Example:** IAM + SSO + Zero Trust on top of legacy AD/LDAP system with a dozen of applications you can't mostly update.





# Recap

# What is security architecture?

Combination of security decisions, which makes actual system's risks manageable in a chosen manner, efficiently, while maintaining all other quality attributes of a system on acceptable level.

# What is security architecture? **TL;DR:**

Set of high-level decisions that simplify security choices, yet drive it in the right direction in coordinated way.

# How to design a security architecture?

- Risk management: SA + S
- Attack surface management: SA + M + S
- Balance tradeoffs: M + S

# How to design a security architecture?

- Risk management: Design against risks
- Attack surface management: Choose your battles wisely
- Balance tradeoffs: Remove conflicts

# How to design a security architecture?

- Risk management: Business, tech decisions
- Attack surface management: Tech, architecture decisions
- Balance tradeoffs: Architecture decisions

There are various directions for security improvement:

- Improve risk management / risk posture.
- Add security controls and tools.

Security architecture enables **systematic risk treatment** that is informed by both to make implementation fit both engineering and business FRs and NFRs.



# Thank you!

[cossacklabs.com](https://cossacklabs.com) / [ivy.chapel.in](https://ivy.chapel.in) /  9gunpi