RSAConference2015
San Francisco | April 20-24 | Moscone Center

CHANGE
Challenge today's security thinking

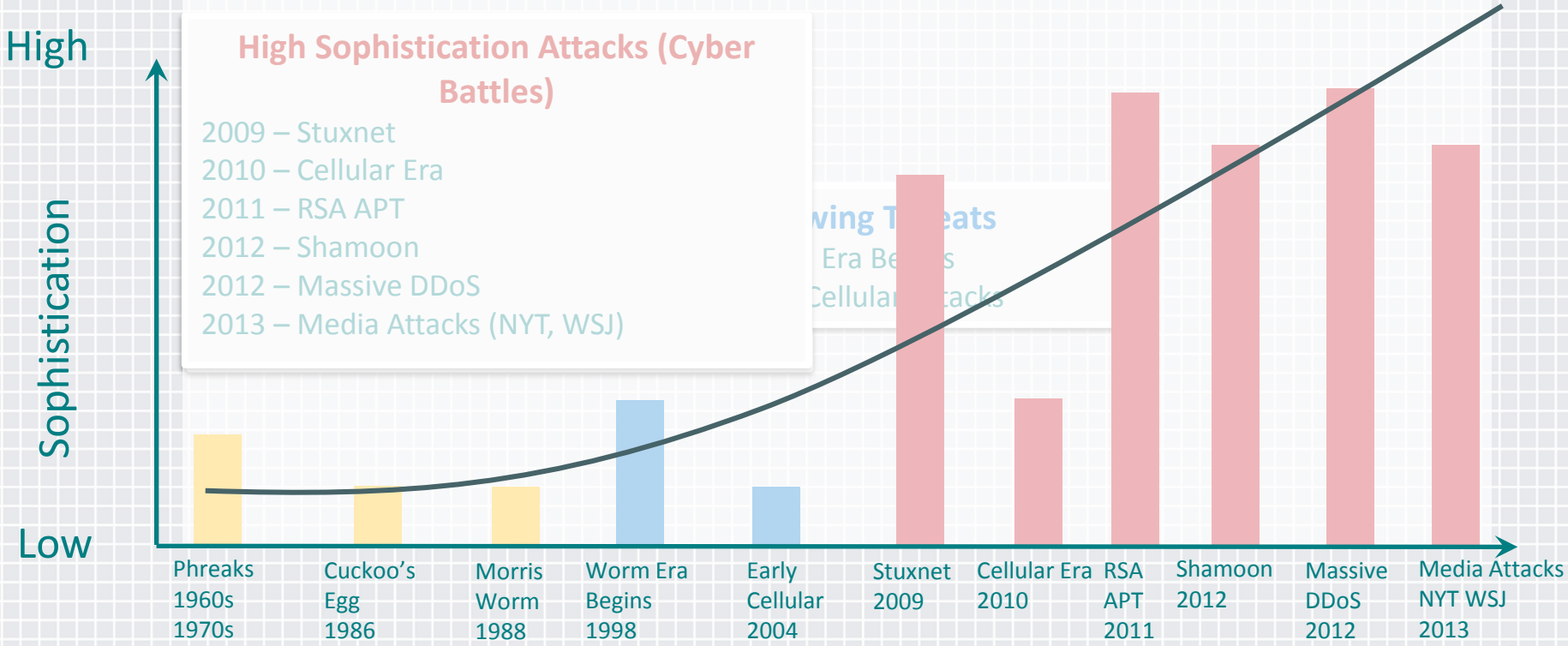
SESSION ID: MBS-T10

Wanted:
Innovation in Security Research

Gus de los Reyes, PhD

Executive Director
AT&T Security Research Center

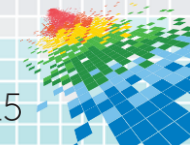
Evolution of Security Attacks



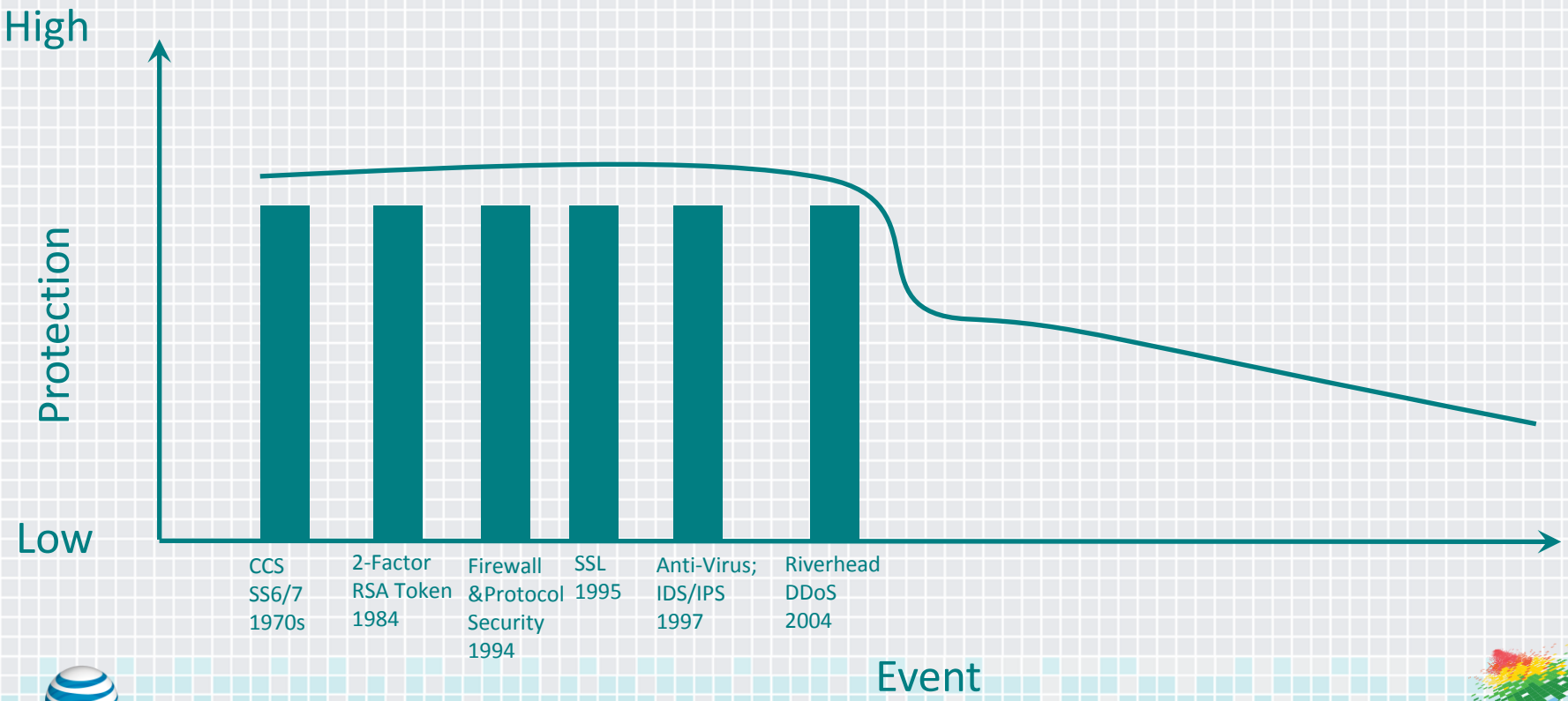
The Top 10

- 2009 Stuxnet
- 2008-2009 Conficker
- 2007-2008 Storm Worm
- 2004 Sasser
- 2003 MyDoom
- 2003 Sobig.F / Nachi(Welchia)
- 2003 Blaster/Lovesan
- 2003 Sapphire/SQL Slammer
- 2001 Code Red II
- 2000 ILOVEYOU

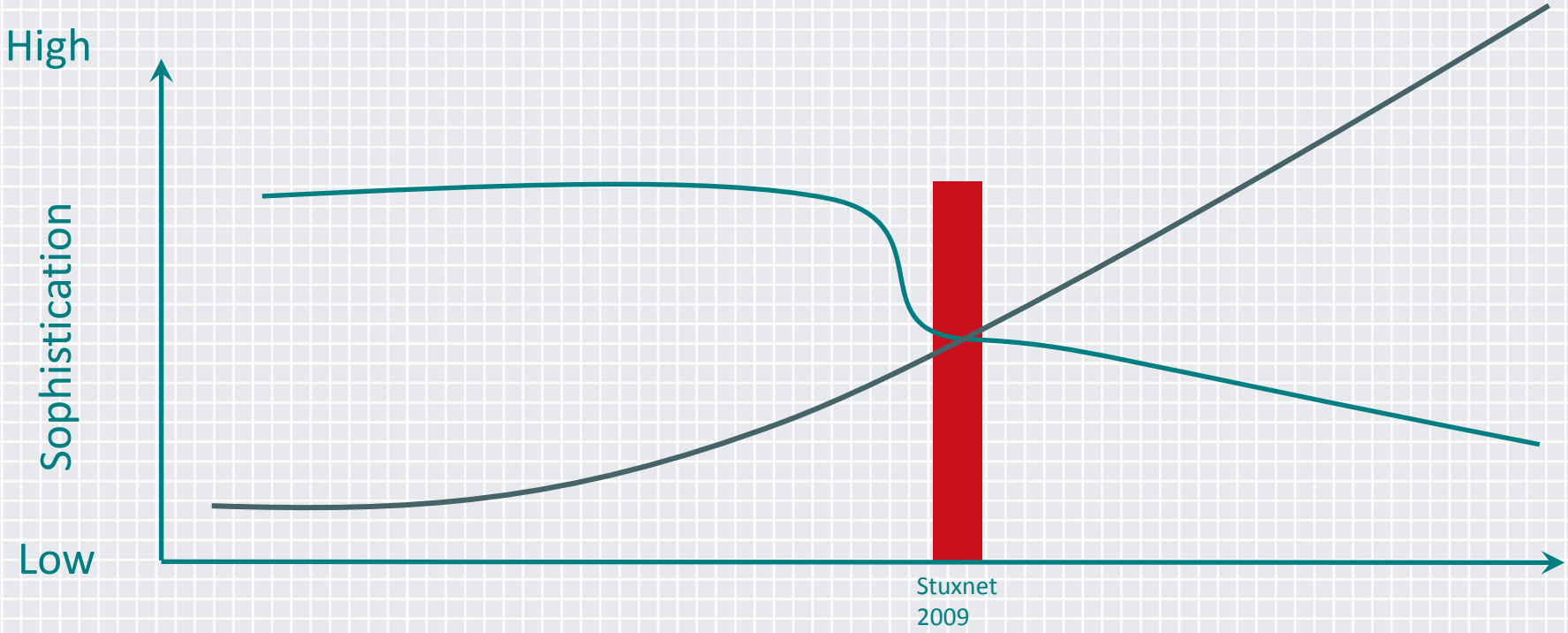
Trojan Horse



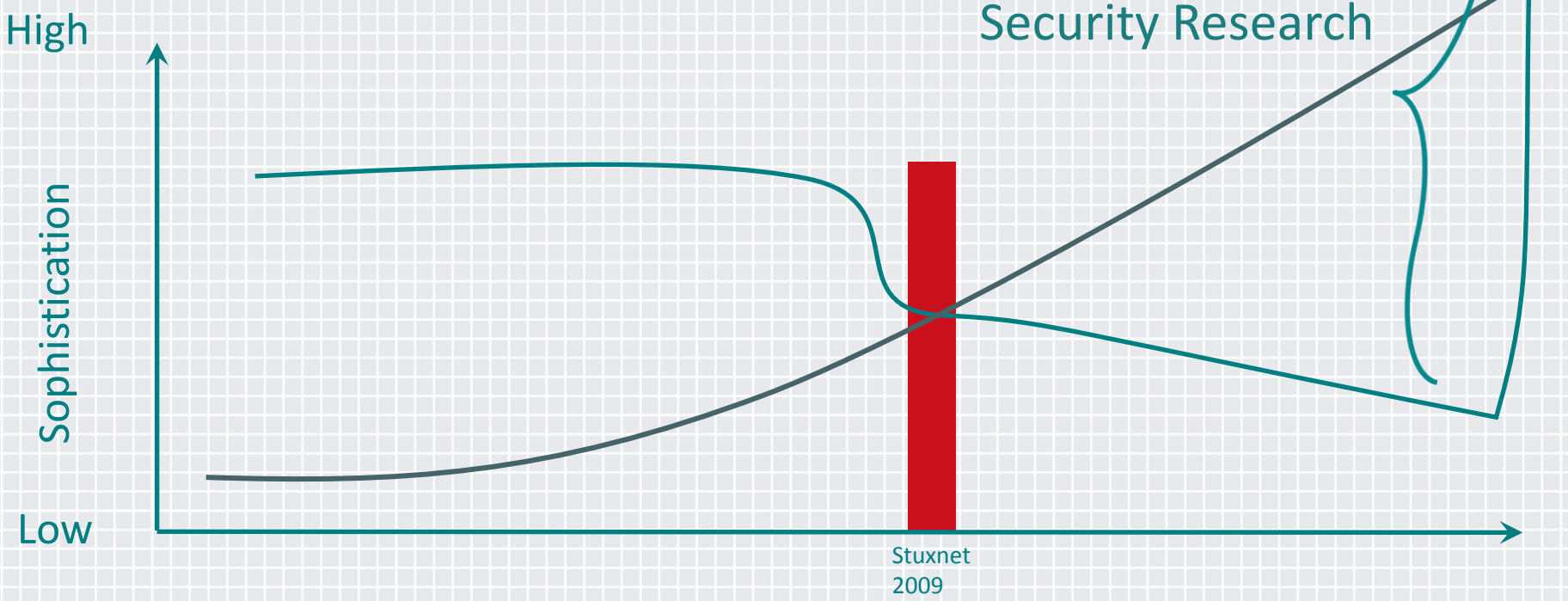
Evolution of Security Defenses



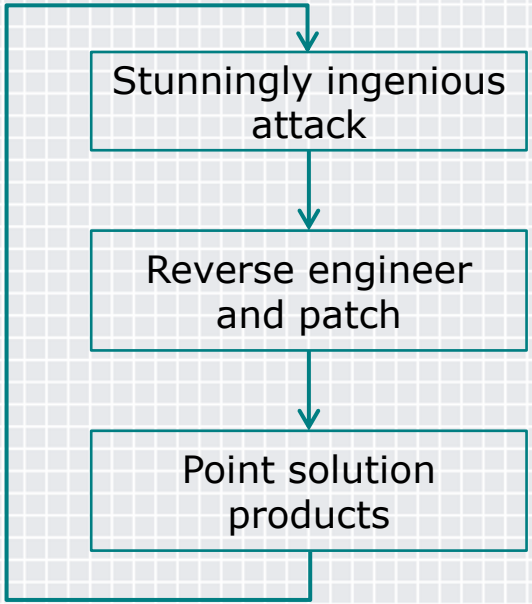
The Crossover Point



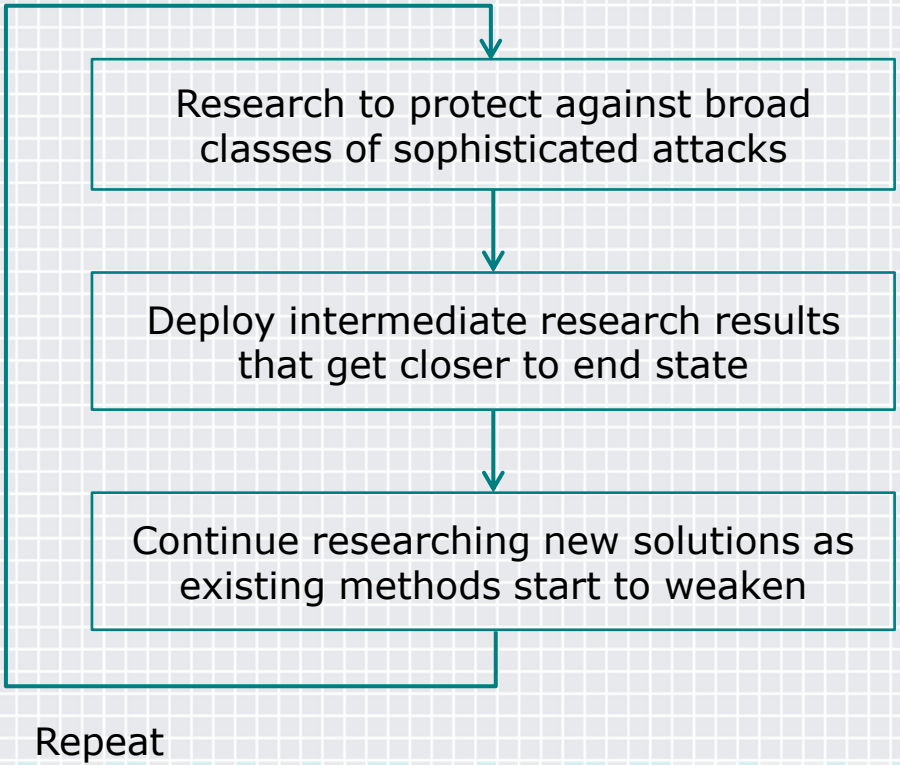
The Need for Security Innovation



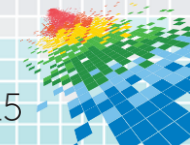
Current Security Cycle



AT&T Security Research Cycle

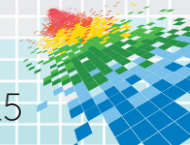


How do we get started?



Security Research Vision

- Out-innovate attackers

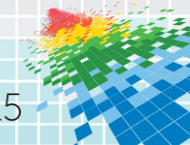


Security Research Goals

- 100% Mobility Availability against Security Attacks
  - Protect mobility as we connect billions of devices
- Unquestionable Cloud Security
  - Secure the cloud to host anything from the virtual enterprise to virtualized networks
- Deep Learning for Network Security
  - Squeeze all of the security potential from our network

Approach

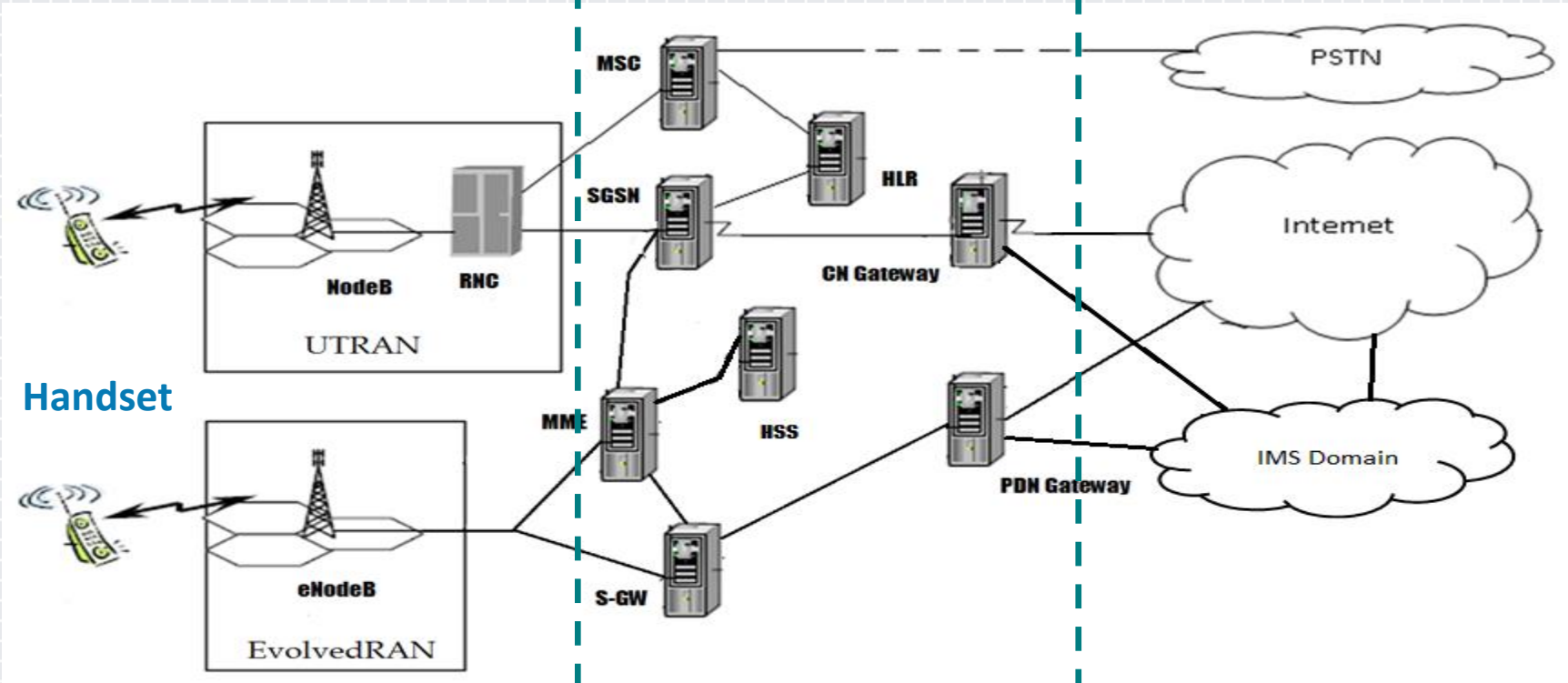
- Look at the end-to-end system to be protected
- Extract features that are common across attacks
- Come up with a solution mosaic
- Fill in mosaic with available tiles
- Complete mosaic with R&D



Business Value of Success

- Maintain the value of Intellectual Property
- Provide extreme flexibility in running a business
- Wring the maximum potential from new technology

Mobility Network



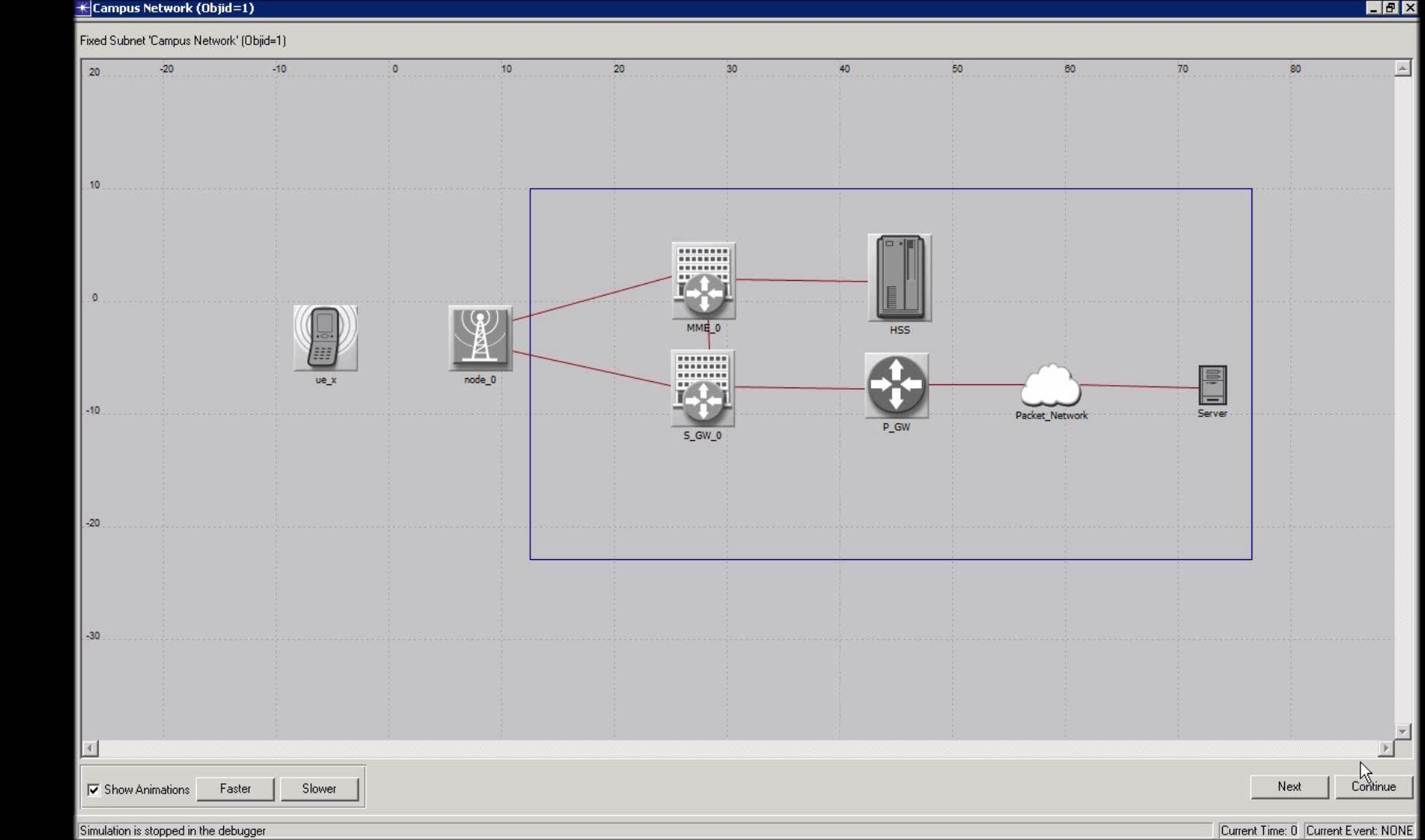
Mobility Attack Taxonomy

	Local	Core (EPC)	Internet
Low traffic load (DoS)	• Jamming • Base station SW vulnerabilities	• Core network SW vulnerabilities (MME, S-GW, etc)	• Spread malware • APT
High traffic load (DDoS)	• Base station saturation • Protocol misbehavior	• HSS saturation • Amplification attacks • Botnet of phones	• Botnet of phones • DDoS attack against Internet nodes affecting network availability
Insider Attack	• Base station shutdown	• Node damage/shut down • HSS saturation • EPC saturation	• DDoS against 3rd party cellular network • 3rd party HSS saturation

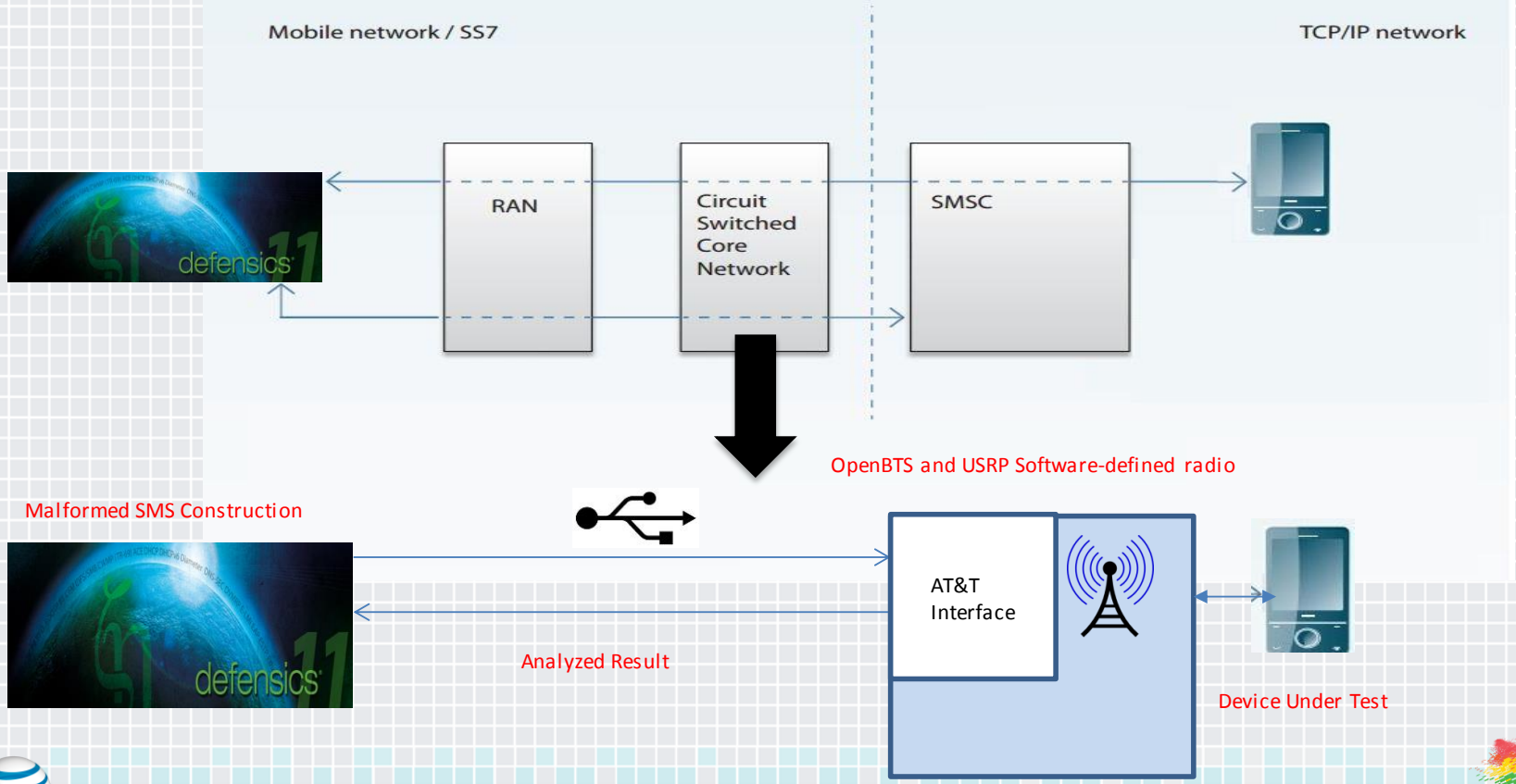
Solution Tiles

	Access network	Core network	IP domain
HTML/Apps	Does not apply	•Titan container •App repackaging detect. •Malware detection	
TCP/UDP/IP	Does not apply	•Anomaly detection •Core network fuzzing •SMS spam detection •M2M security	•Saturn •DDoS protection •IP/DNS reputation •Anomaly and attack detection
MAC/RRC/RLC PHY	•eNodeB and core network fuzzing •Protocol misbehavior detection •Low layer network security •Network attack detection and mitigation •Fake base station defenses		Does not apply

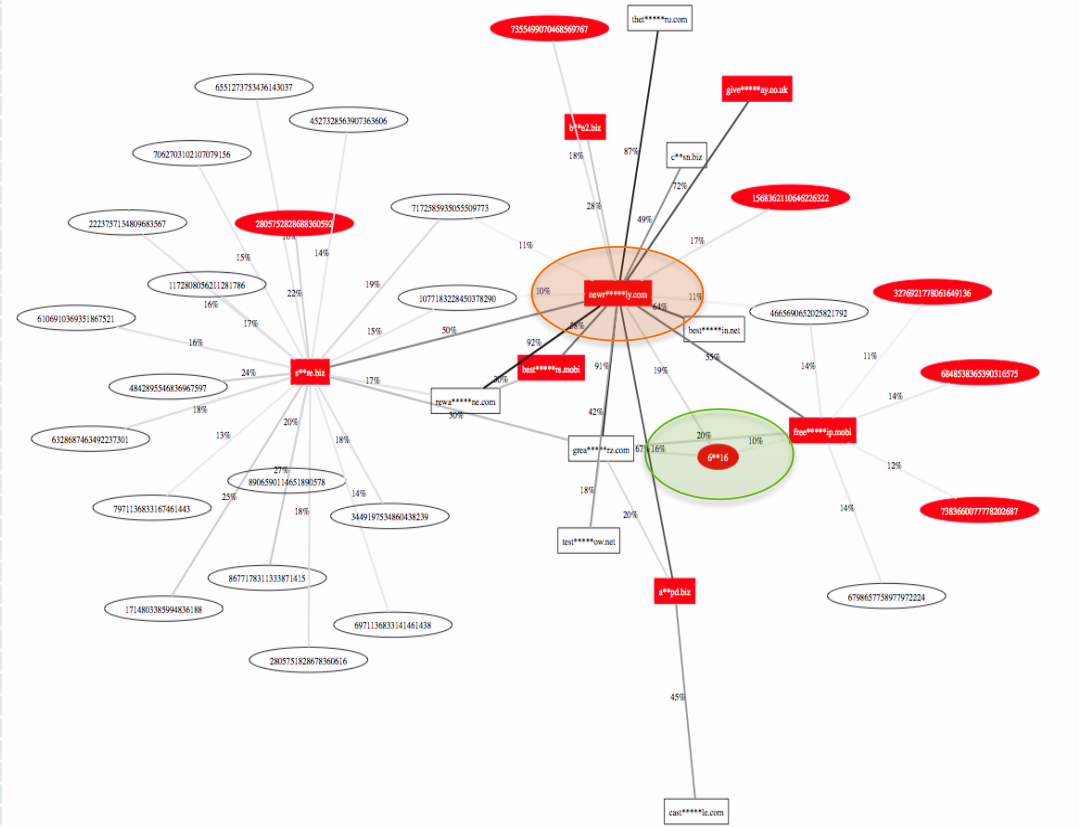
Handset → Titan container, malware detection/mitigation, handset SMS fuzzing, provably secure transactions.


Campus Network (Objid=1)
Fixed Subnet 'Campus Network' (Objid=1)
20
-20
-10
0
10
20
30
40
50
60
70
80
10
0
-10
-20
-30
ue_x
node_0
MME_0
HSS
S_GW_0
P_GW
Packet_Network
Server
Show Animations
Faster
Slower
Next
Continue
Simulation is stopped in the debugger
Current Time: 0
Current Event: NONE

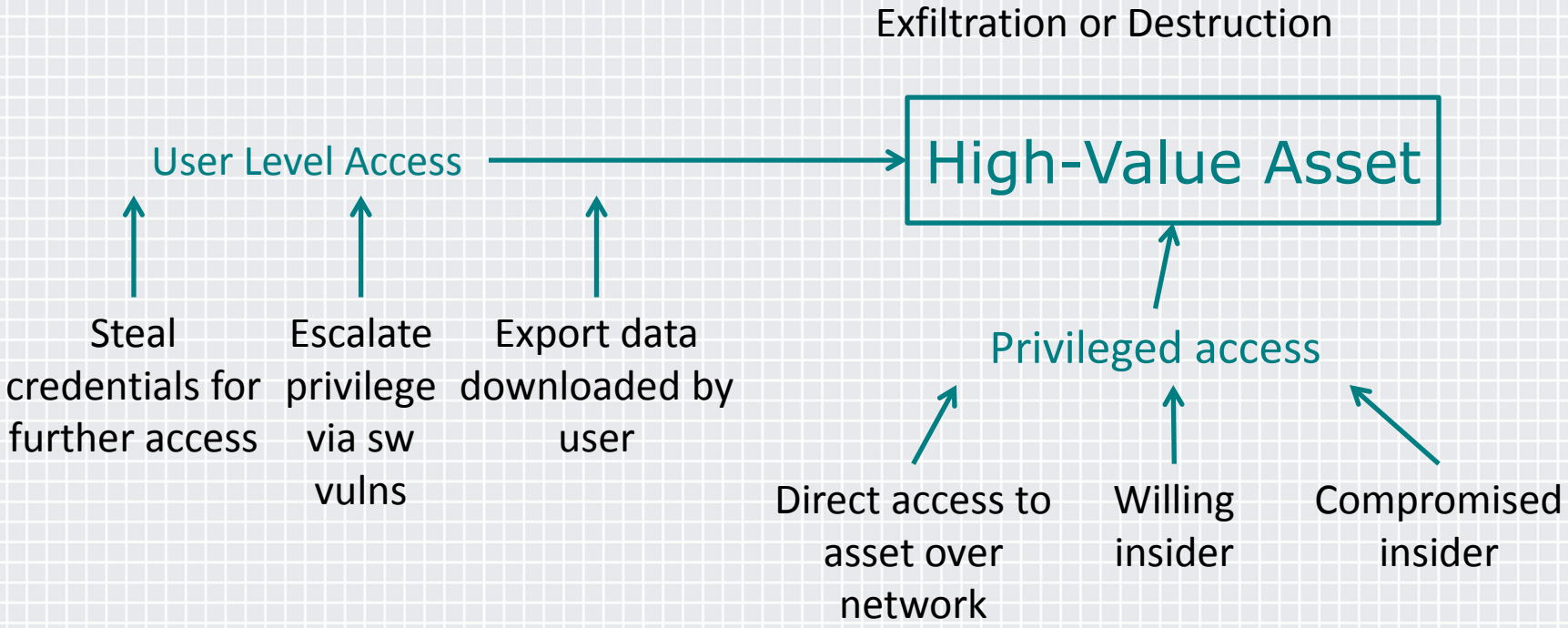
Automated Smartphone Fuzz Testing



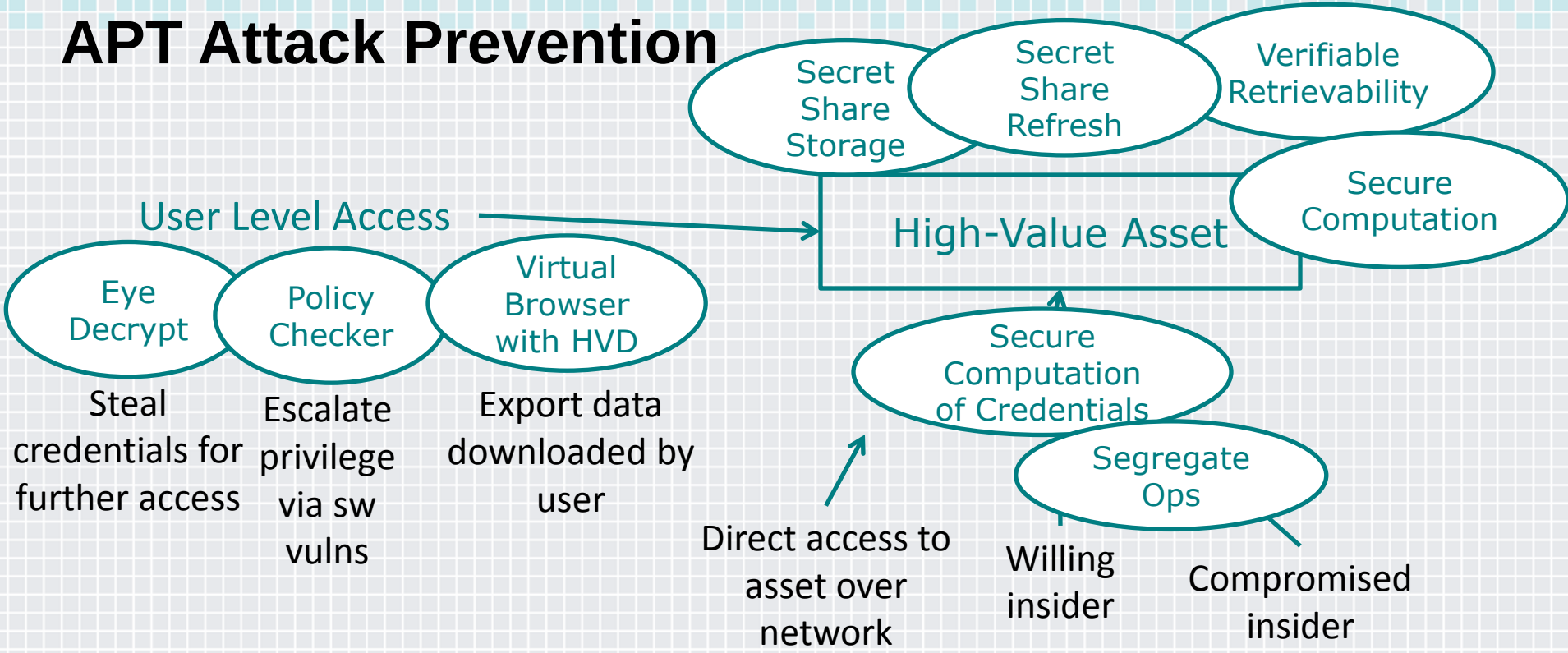
Detect Widespread Attack Campaigns



APT Attack Model

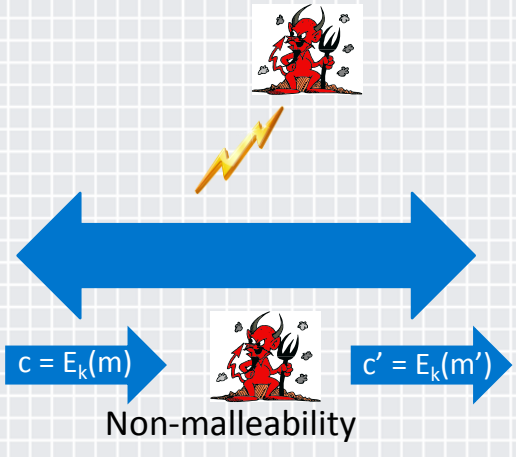


APT Attack Prevention



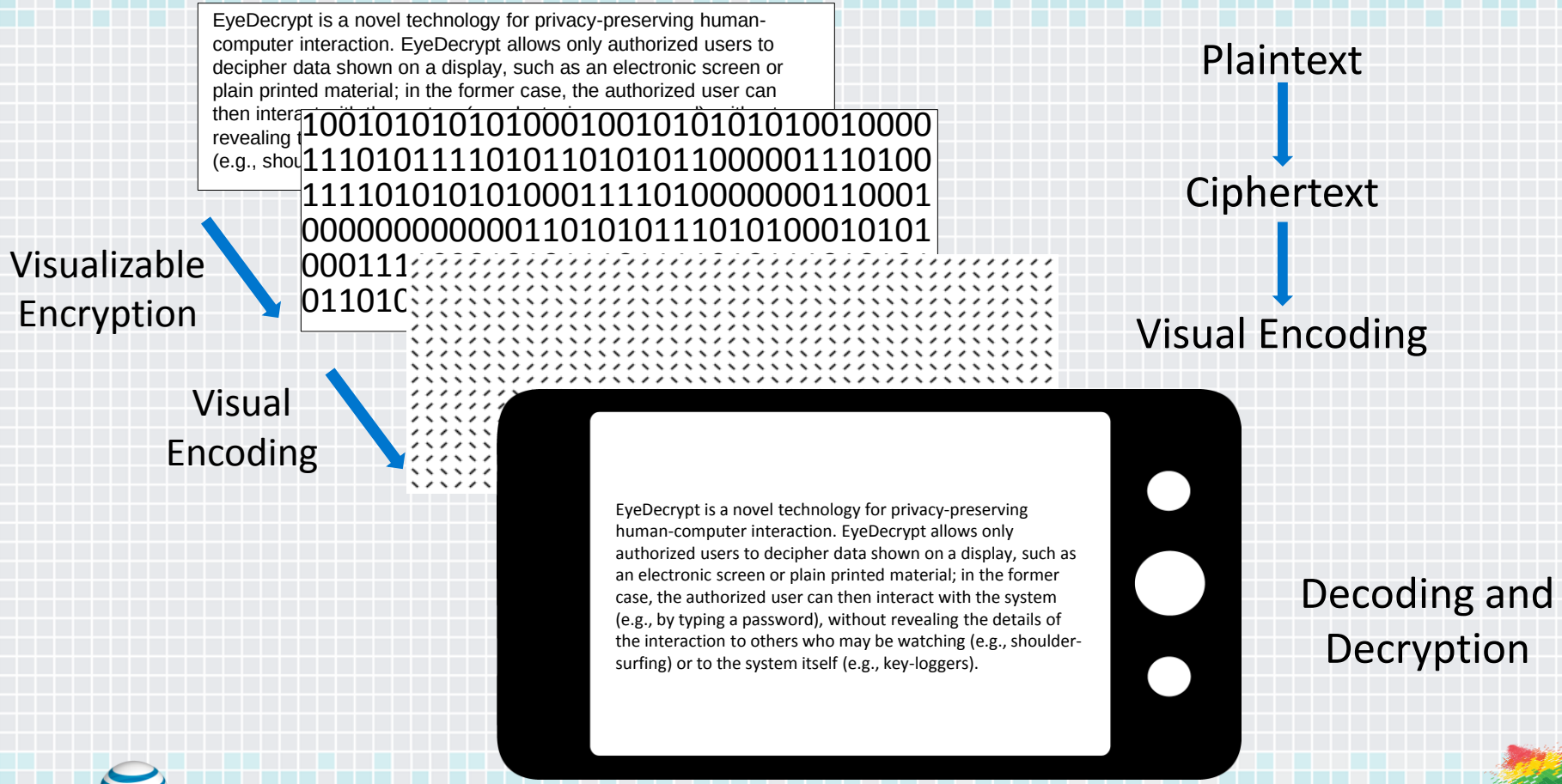
System View

Encrypted Content repository

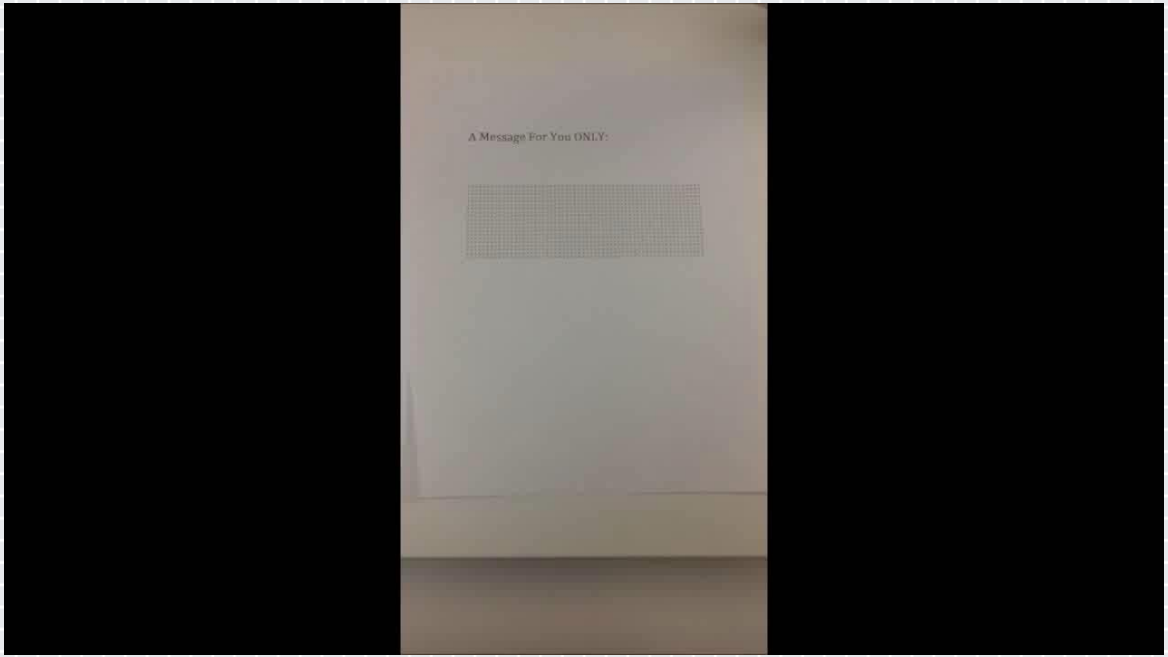


(Public) Rendering/input device

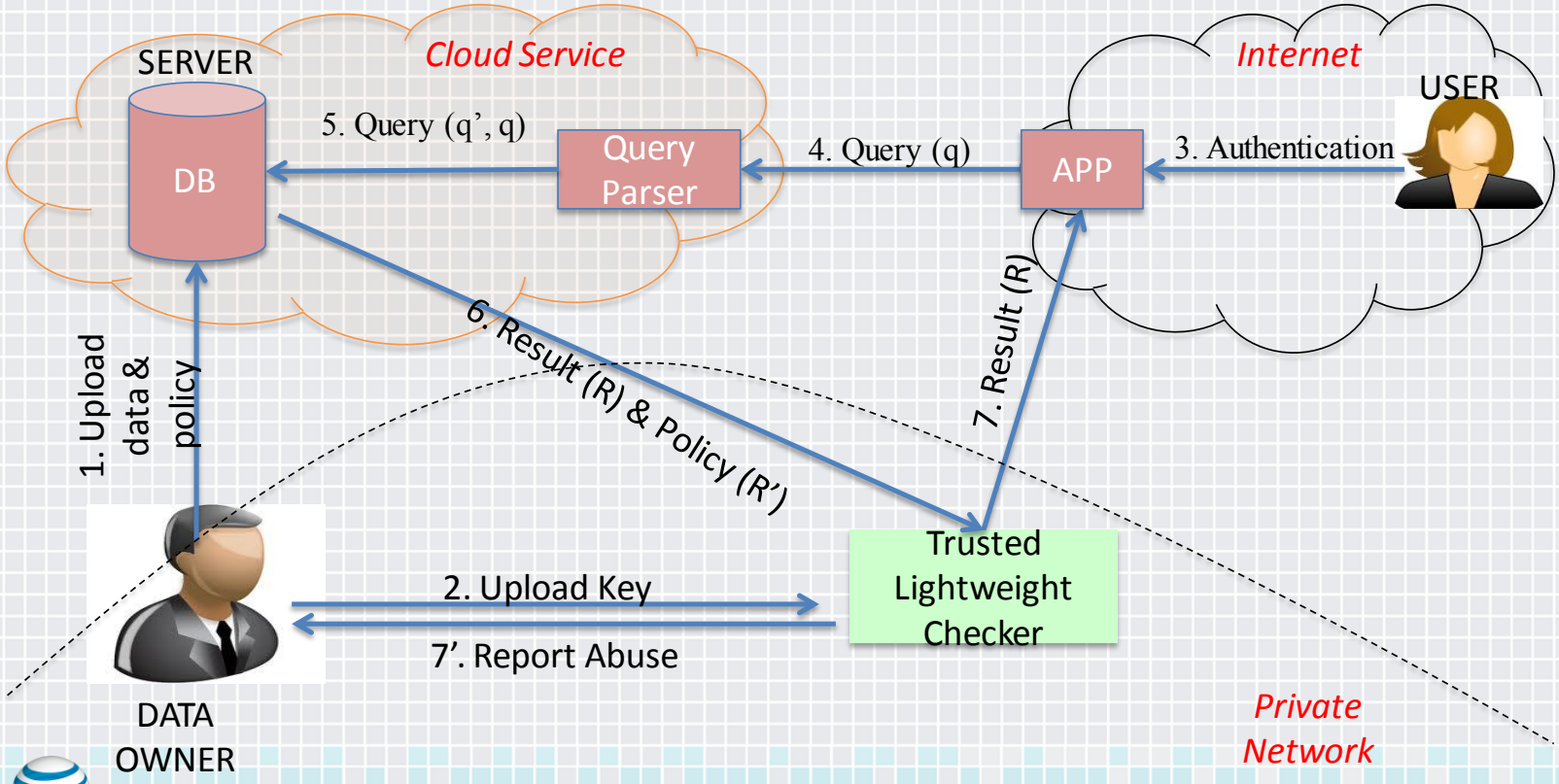
Trusted Device



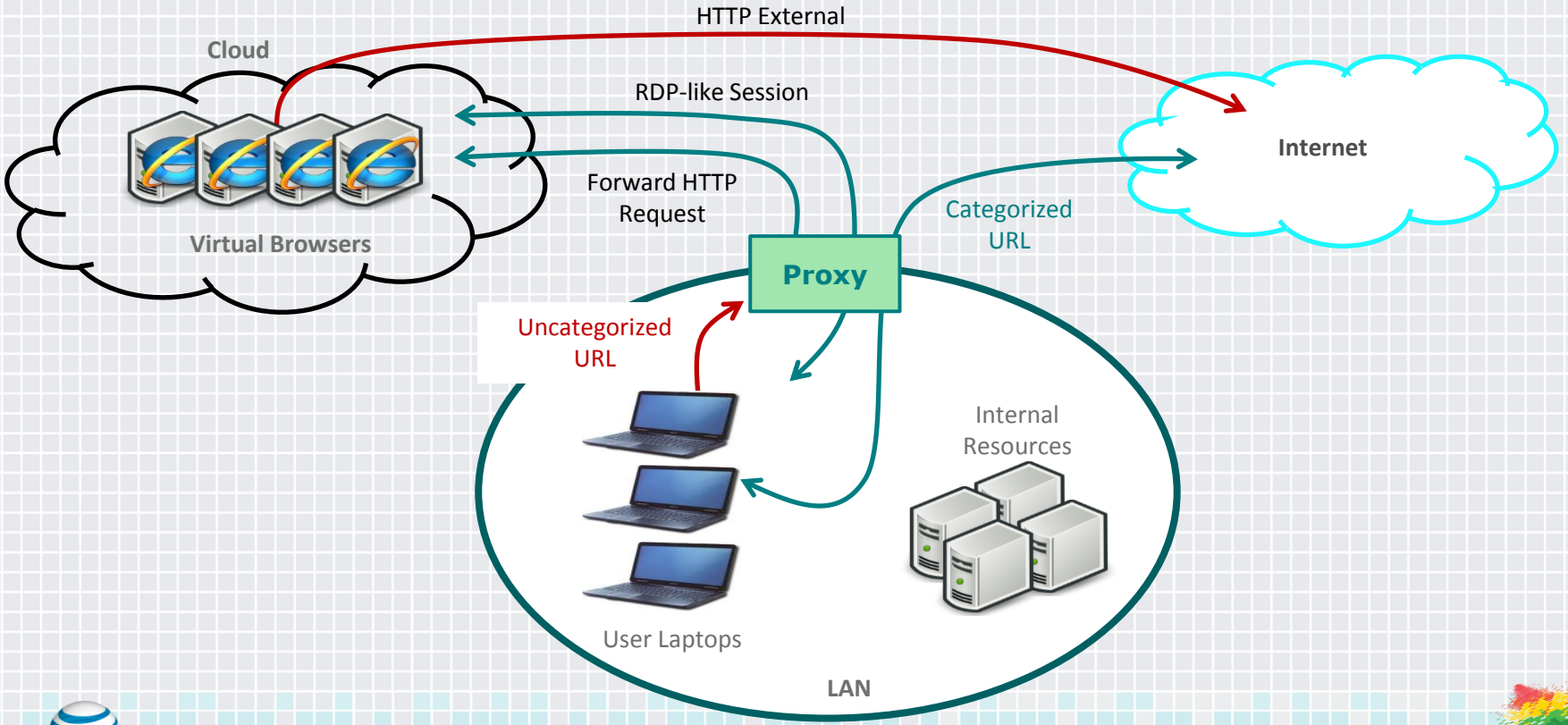
EyeDecrypt Demonstration



Data Access Policy Checker

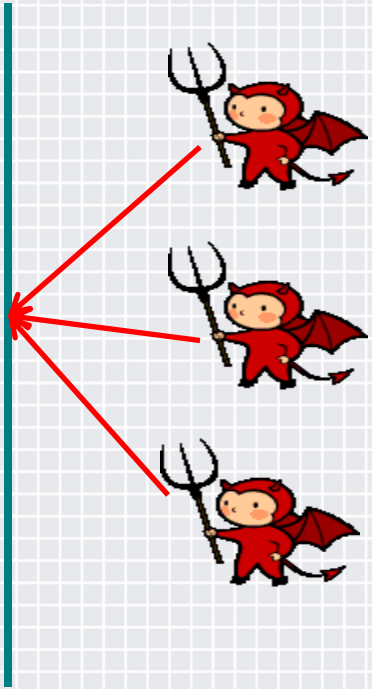


Virtual Network Browser



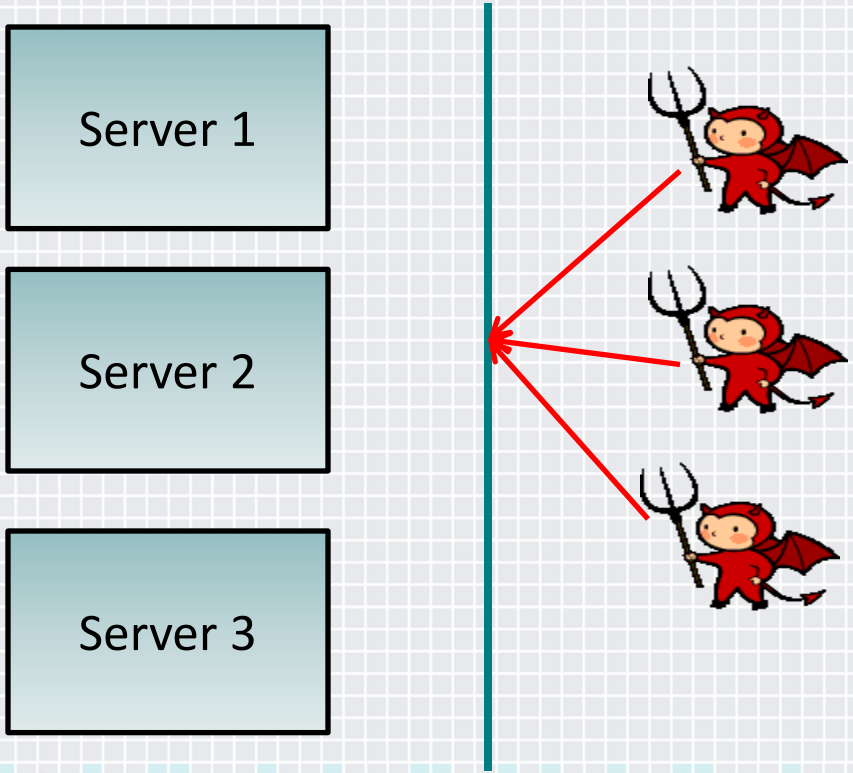
Secure Storage

Servers with confidential data



If adversaries break through the perimeter, They learn confidential information

Secure Distributed Storage



If adversaries break through the perimeter into one server, they learns nothing.

In order to learn anything, adversaries must break into all servers.

A Secure Secret Sharing Scheme

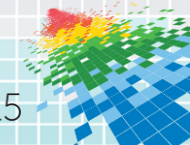
Suppose we have a secret s

An n-out-of-n secret sharing scheme:

Pick n-1 random values $s_1, s_2, \ldots, s_{n-1}$ and set $s_n = s_1 \oplus s_2 \oplus \ldots\ldots\ldots \oplus s_n$

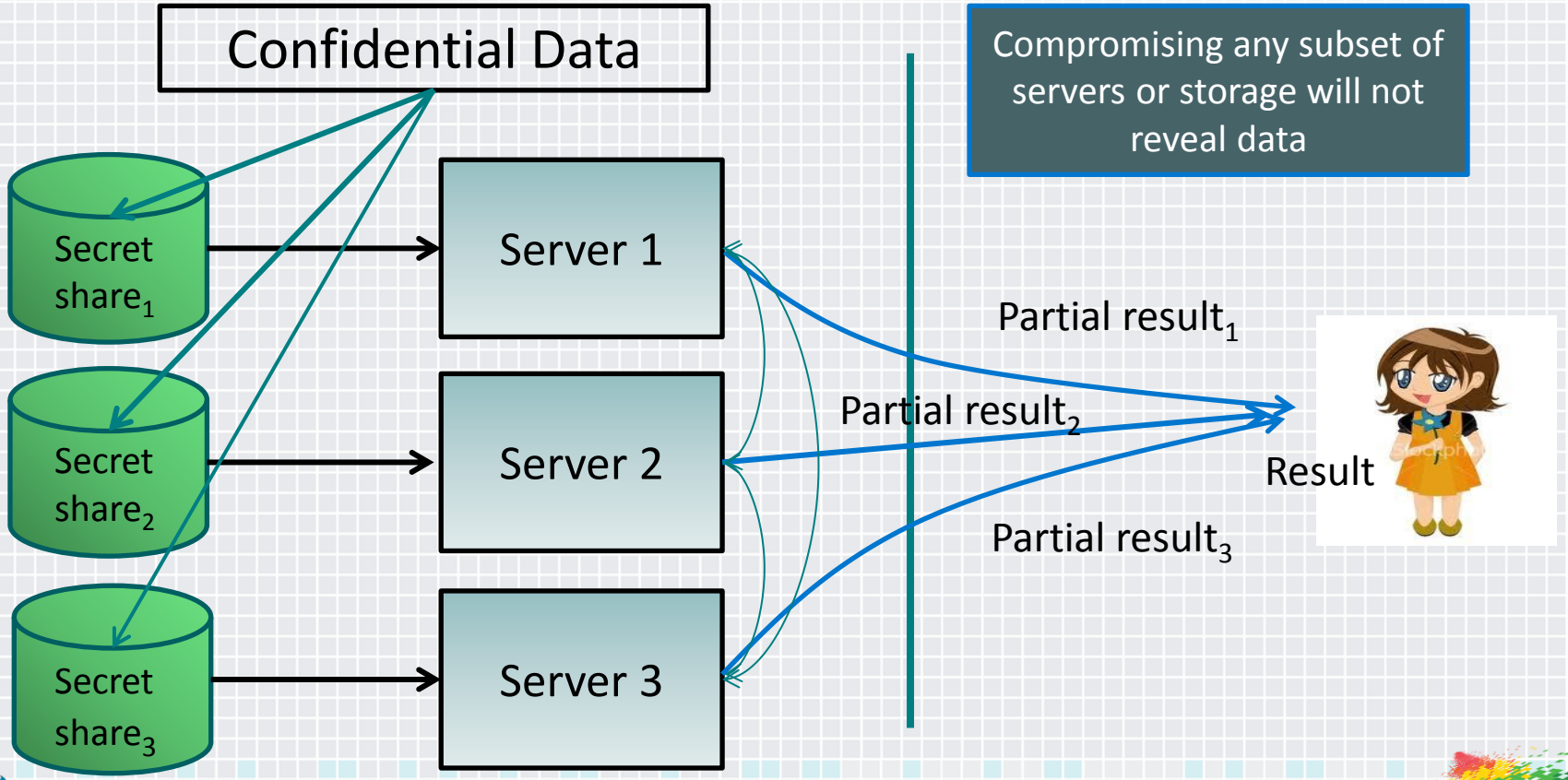
Now, if we store each s_i on a different server, then even if an adversary learns any n-1 of the s_i values, he (provably) learns nothing about s

This can be generalized to k-out-of-n secret sharing scheme such that even if an adv learns k of the s_i values, he learns nothing about s; Advantage is that s can be reconstructed using any k+1 out of the s_i values (better reliability)



Secret Sharing and Secure Computation

#RSAC



Rethink Possible