File Integrity Monitoring for z

Kindly Sponsored by
NewEra Software

Presented By:

Al Saurette	(403) 818-8625	al@maintegrity.com
Brandon Saurette	(587) 897-7502	brandon@maintegrity.com

Overview

Current Situation

What is FIM?

How does FIM+ help me?

Demo simple scan

Demo deploy / DevOps

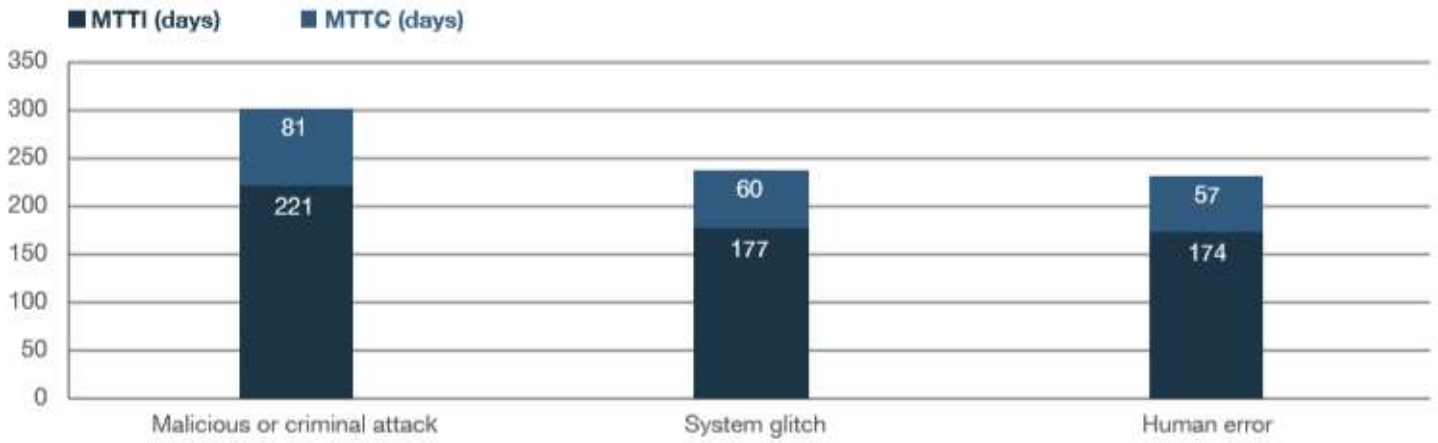
Performance

Q & A

Business Risk – July 2018

- 2018 Ponemon study 477 companies
- Mean time to identify a breach 197 days
- Mean time to contain a breach 69 days

Figure 27. Days to identify and contain data breach incidents by root cause



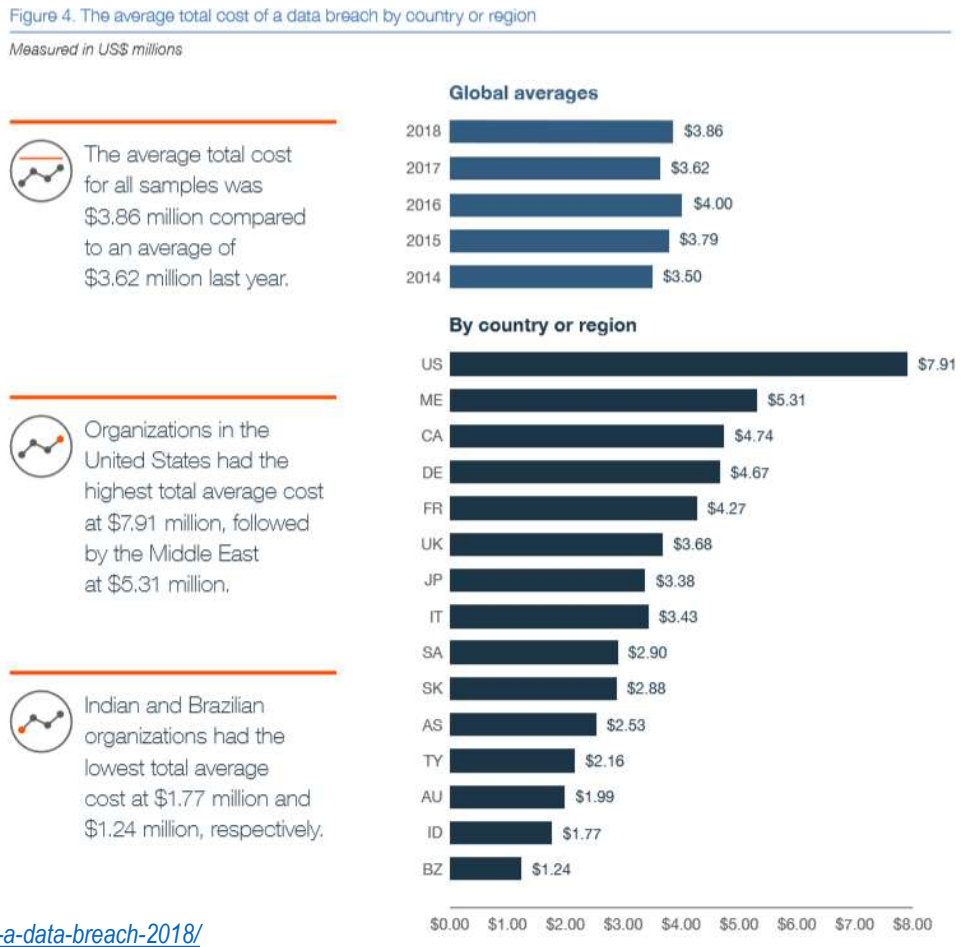
IBM sponsored 2018 Ponemon *Cost of Data Breach Study*
https://securityintelligence.com/ponemon-cost-of-a-data-breach-2018/

Business Risk – July 2018

Why Do I Care?

USA Average cost: $7.91 Million

Unquantifiable brand and reputational impact



IBM sponsored 2018 Ponemon *Cost of Data Breach Study* https://securityintelligence.com/ponemon-cost-of-a-data-breach-2018/

What does FIM do?

- **File Integrity Monitoring (FIM)**
  - Take a snapshot of a file or whole application at trusted level
  - Save keys in the encrypted vault
  - Later scan files / programs in use to detect any alterations
- **FIM+ monitors changes in**
  - Executable programs, source, JCL, config members, panels
  - IMS / DB2 / SMF log files, sequential, PDS, PDSE and encrypted files
  - USS / HSF, Shell scripts, Java, binaries, html
- **Integrates with SIEM** (Splunk, QRadar, et al)
  - Alerts sent to SIEM for standard escalation
  - Focus incident response on right interval (since last success)

FIM+ is MainTegrity's product for z/OS

How does FIM+ help me?

Intrusion Detection:

Identify internal & external attacks that bypass access control
Identify altered, added and deleted modules
Reduce MTTI / MTTC from 197 + 60 days to minutes[1]

On-demand Integrity Validation

Bit by bit clarity that components in use match approved versions
Satisfy immediate Audit or management request for confirmation

Compliance:

Automated FIM required for new PCI, NIST standards
Success records prove regular checking
Save real $$$ by reducing the time & effort spent on audits
Give audit what they want and get back to work faster

[1] IBM sponsored 2017 Ponemon Institute *Cost of Data Breach Study: Global Overview* - https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=SEL03130WWEN

Compliance

PCI DSS (3.2)

- ✓ 10.5 - Use file-integrity monitoring or change-detection software on logs to ensure that existing log data cannot be changed without generating alerts (although new data being added should not cause an alert).
- ✓ 11.5 - Deploy a change-detection mechanism (for example, file-integrity monitoring tools) to alert personnel to unauthorized modification (including changes, additions, and deletions) of critical system files, configuration files, or content files; and configure the software to perform critical file comparisons at least weekly.

NIST

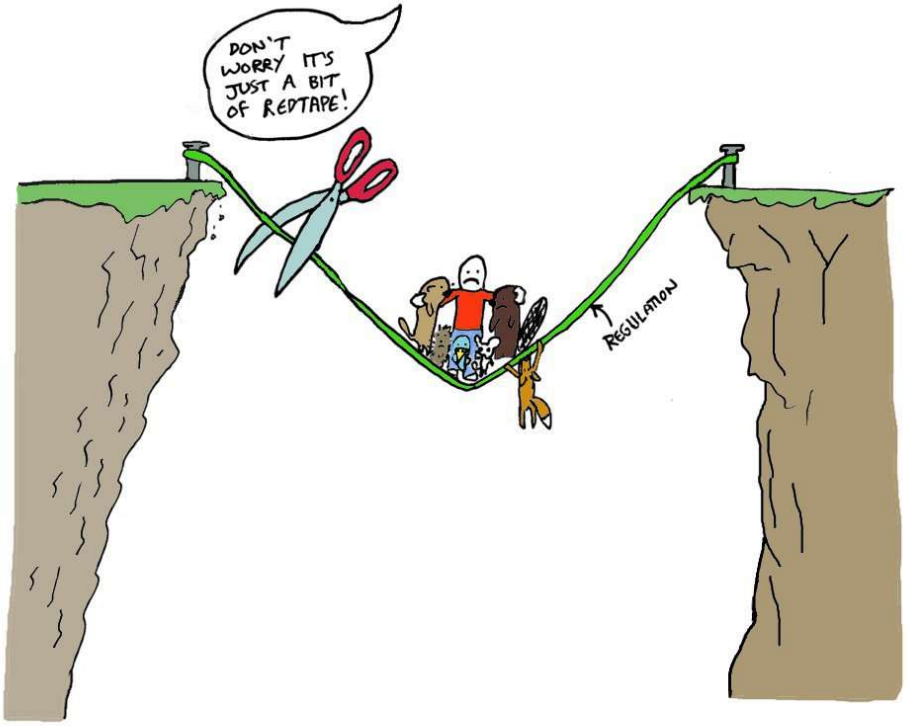
- ✓ SP 800-53 (FISMA): Control SI-7 “the organization employs integrity verification tools to detect unauthorized changes to [Assignment: organization-defined software, firmware, and information].”
- ✓ SP 800-66 (HIPAA): Implement policies and procedures to protect electronic protected health information from improper alteration or destruction.

FPS-140

- ✓ A cryptographic module shall perform the following power-up tests: cryptographic algorithm test, software/firmware integrity test, and critical functions test

GDPR

- ✓ Article 32 – Security of Processing
  - (b) “ensure the ongoing confidentiality, integrity, availability and resilience of processing systems and services.
  - (d) process for regularly testing, assessing and evaluating effectiveness of technical and organizational measures



Regulation protects the things we care about - we need to keep it that way

@CARTOONRALPH

How does FIM help me more?

Finally Deal with Known Security Exposures

SMPE Injection

Production Drift:

QA diverges from Prod - emergency changes

Retroactive correction of old problems

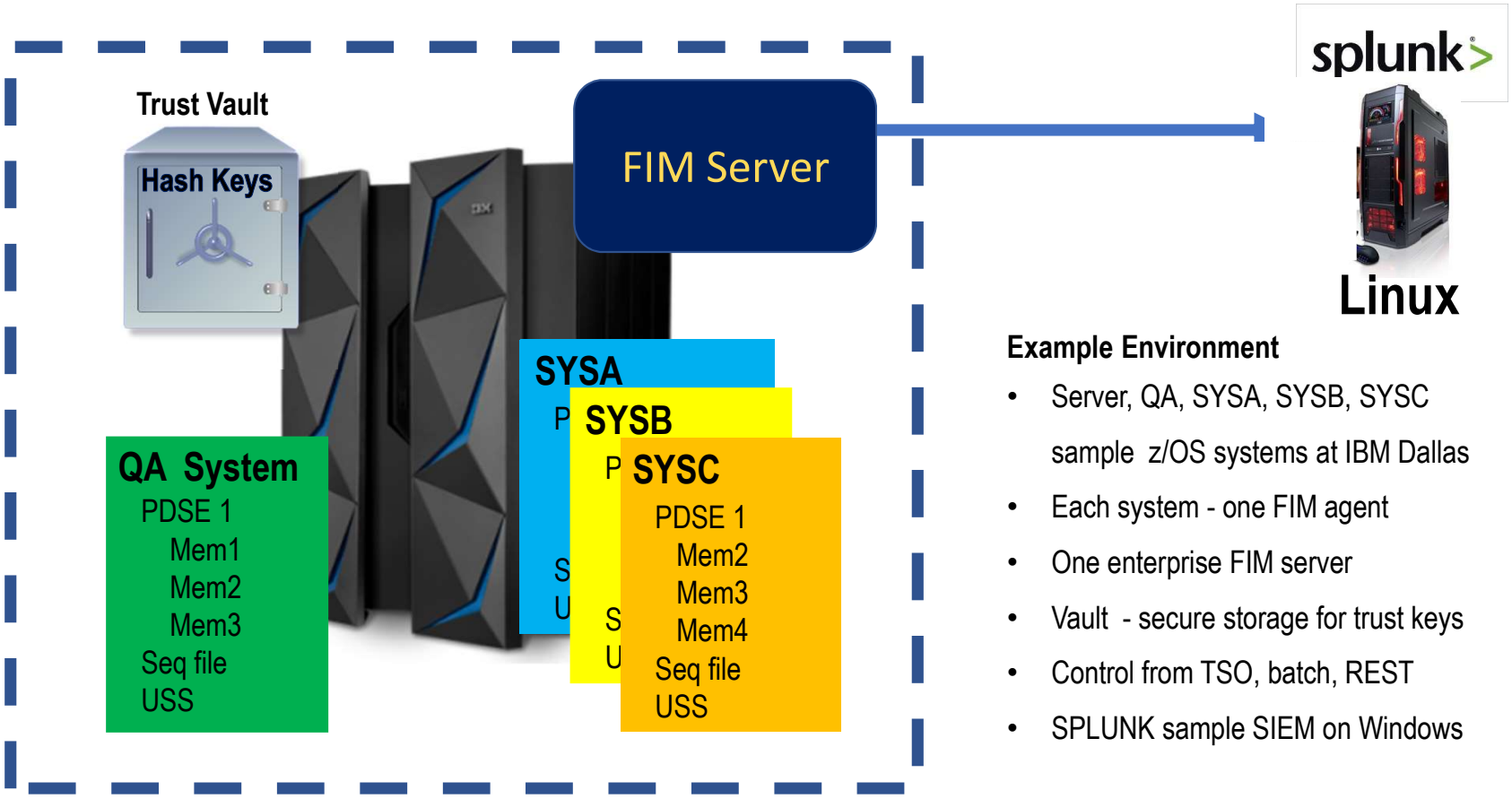
Deploy Audit:

Positive confirmation deploy was successful

Wrong version, missed, changed, added modules detected

Monitor at file and group levels

Sample Environment



Example Environment

- Server, QA, SYSA, SYSB, SYSC sample z/OS systems at IBM Dallas
- Each system - one FIM agent
- One enterprise FIM server
- Vault - secure storage for trust keys
- Control from TSO, batch, REST
- SPLUNK sample SIEM on Windows

Initial Scan

Scan

FIM Server

Baseline Saved

FIM Agent

Prod - SYSA

Sys1.ProdLib
- Mem1
- Mem2
- Mem3

Seq file

USS

Result Log

SPLUNK>

Validation Scan

Validate

FIM Server

Validate Success

Result Log

Baseline Saved

SPLUNK>

Baseline Saved

FIM Agent

Prod - SYSA

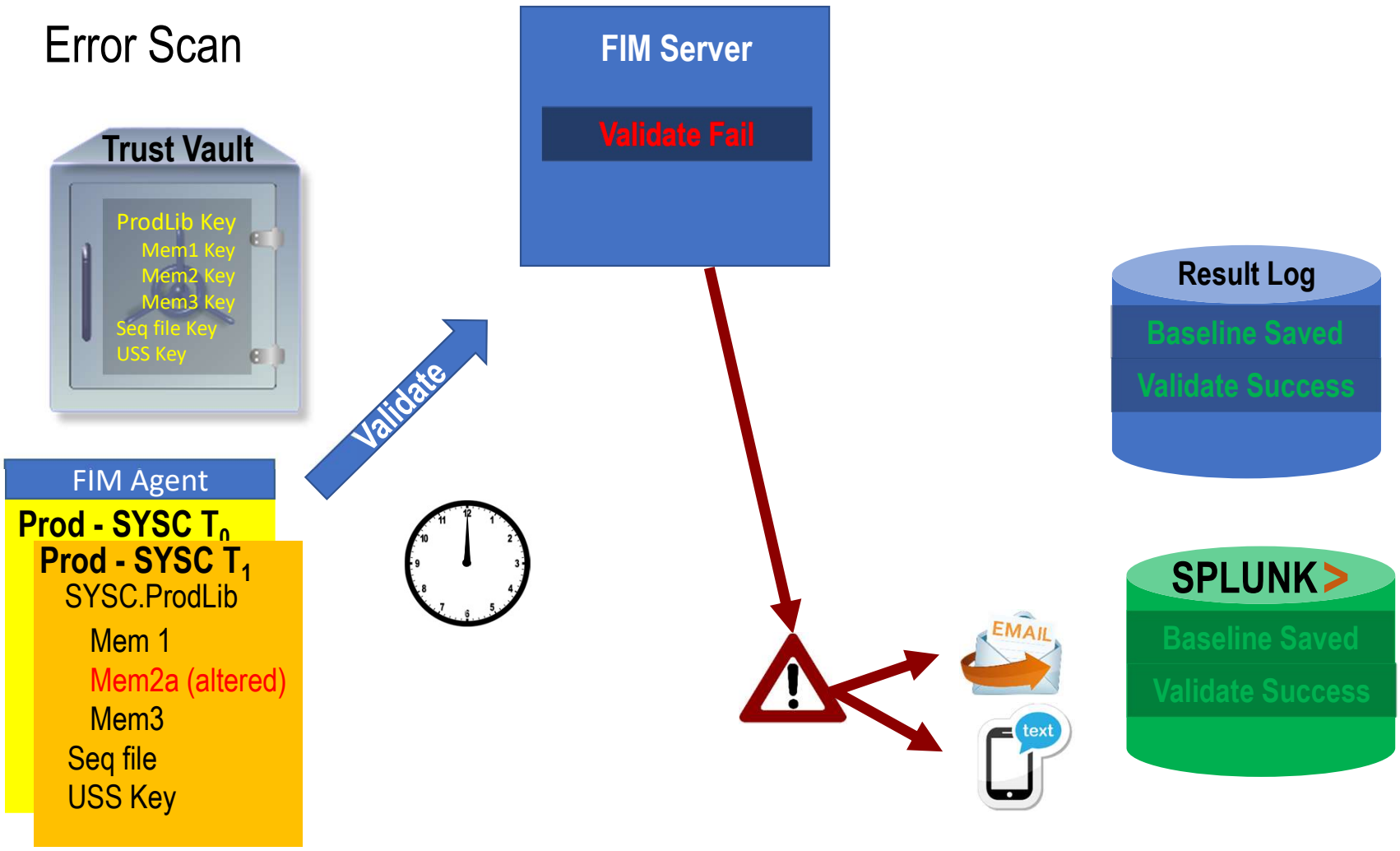
Sys1.ProdLib
- Mem1
- Mem2
- Mem3

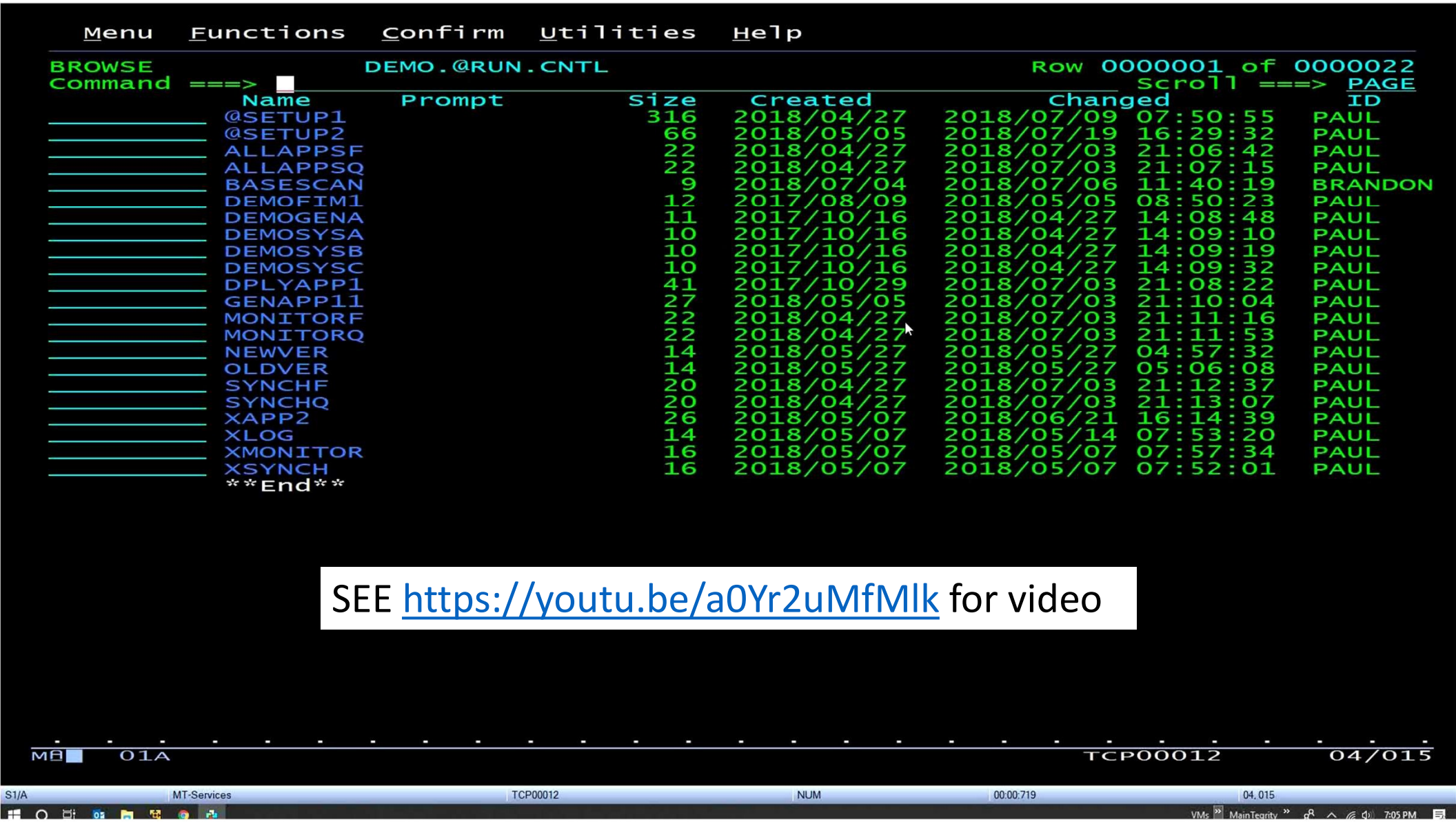
Seq file

USS

SEE https://youtu.be/qbjHb70MVEE for video

Error Scan


FIM Server
Validate Fail
Trust Vault
ProdLib Key
Mem1 Key
Mem2 Key
Mem3 Key
Seq file Key
USS Key
Validate
FIM Agent
Prod - SYSC T0
Prod - SYSC T1
SYSC.ProdLib
Mem 1
Mem2a (altered)
Mem3
Seq file
USS Key
Result Log
Baseline Saved
Validate Success
SPLUNK>
Baseline Saved
Validate Success
EMAIL
text



SEE https://youtu.be/a0Yr2uMfMlk for video

Example: SMP/E – Typical Install Process

1. SYSMOD copied to a USS directory, decompressed & possibly copied over to z/OS
2. RECIEVE command is issued through SMPE
3. Eventually someone will install the package

Simple…Right?

But how long does it take for someone to get around to installing?

- This Week? This Year?

In this time someone could have slipped a backdoor into the package

SMP/E Injection – Why is this so bad?

Dangerous

- SYSMODS are applied with the permissions granted to SMP/E
- Known z/OS integrity exposure *

Not a sophisticated attack

- Unix System Services / Jar files nothing new to Unix hackers…even if z/OS is
- Free information and exploit frameworks exist on the open web
- Often, SYSMODS have weak file permissions allowing for modification

* Chad Rikansrud / Mark Wilson - SMP/E Abused, Share 2018
https://share.confex.com/data/handout/share/130/Session_21903_handout_11399_0.pdf -->

SMP/E Injection – Mitigation

Use FIM+ to scan a SYSMOD

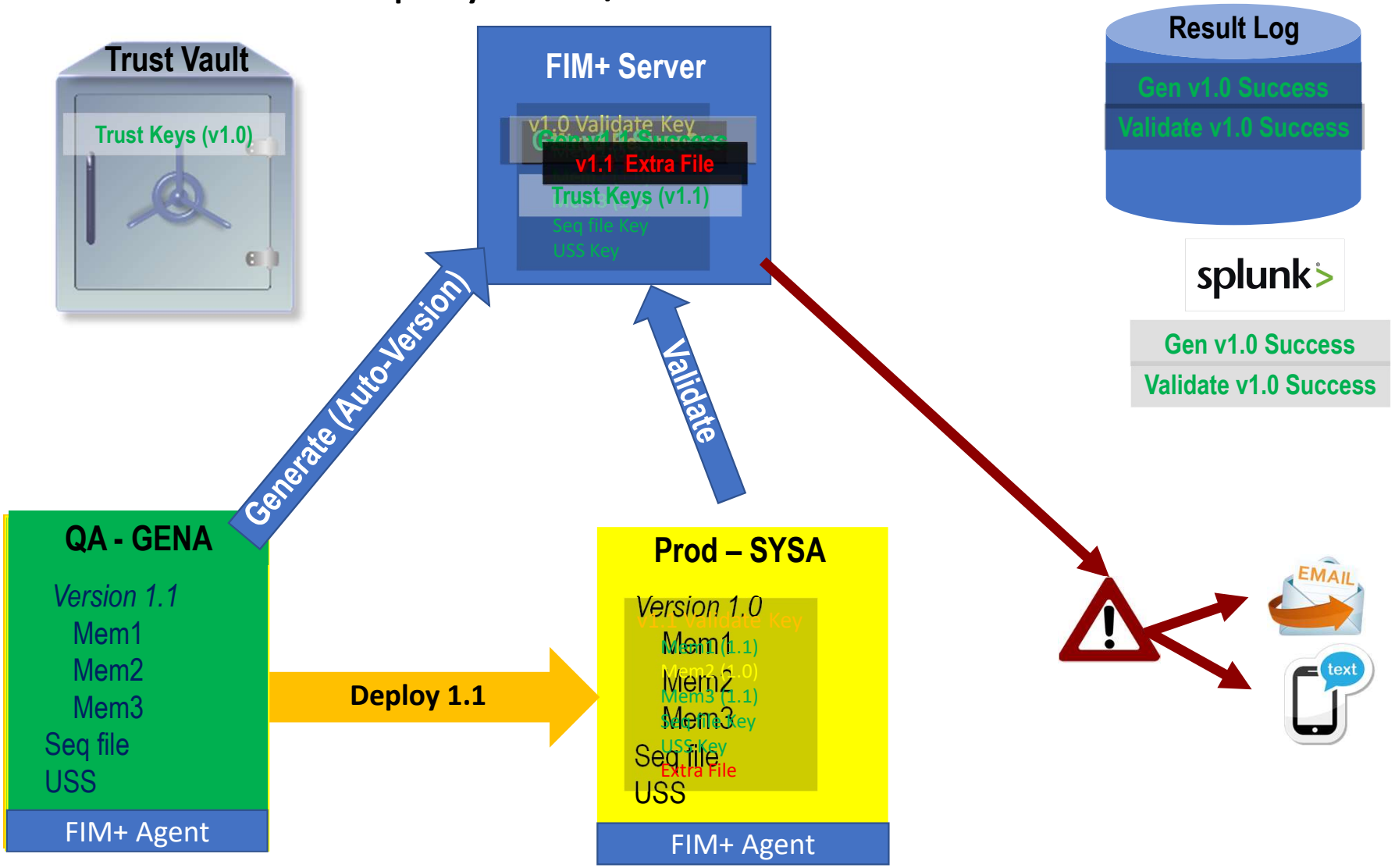
- Scan .gzip as received from the vendor
- Scan mods when extracted
- Re-scan mods prior to installation

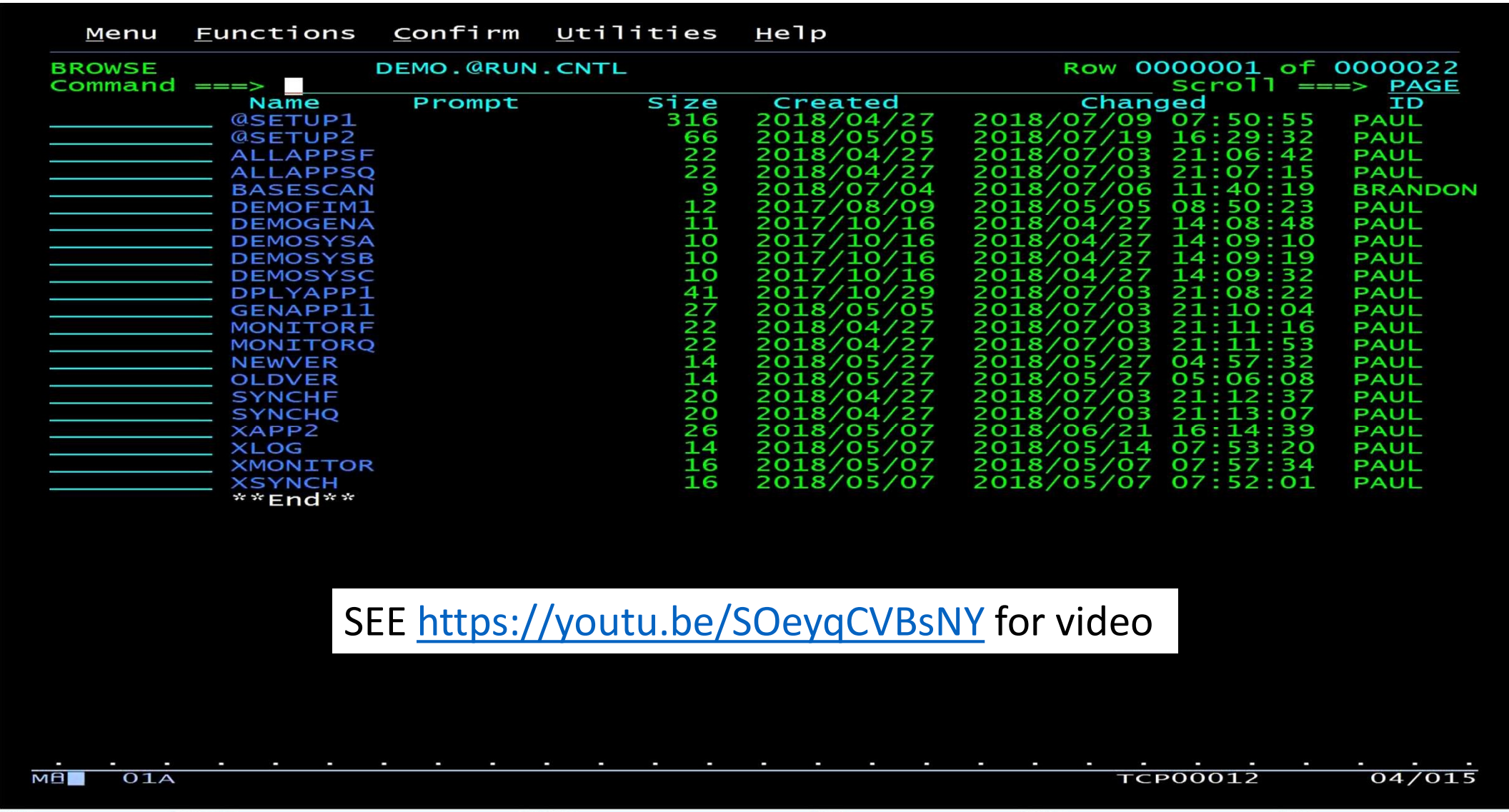
Effective

- By comparing validation keys with the trust keys in the vault, altered packages can be detected before installing them.

Lobby your software vendors to provide a SHA-256 standard key

Deployment/Validation of V 1.1





SEE https://youtu.be/SOeyqCVBsNY for video

How long does it take?

Scan sample Sys1.Linklib (4162 modules)

- Quick scan: **< 0.01 sec CPU,** 1 second elapsed
  1 million modules - about 2 CPU seconds
- Full Scan: **< 2 sec CPU, < 1** minute elapsed
  Uses z hardware assist – Crypto / Hashing

Scan whole APF list (149 Datasets, 42,600 members)

- Quick scan: **1 sec CPU,** 15 seconds elapsed
- Full Scan: **36 sec CPU,** 4 minutes elapsed

Plan 1 hour, Install 1 hour, results 1 hour

Quick scans anytime, Full Scans at night - CPU impact is ZERO

All tests conducted at IBM's Dallas Innovation Center on an EC12 w/ ICSF running z/OS 2.2

Many problems, One solution

Hacking, Errors, Glitches - All involve changes to files

- **Intrusion Detection –** identify, determine scope, focus response
- **On-demand Integrity Validation –** Immediate confirmation
- **Compliance -** faster, easier, complete
- **SMPE injection -** keep USS hackers out
- **Production Drift –** avoid QA vs Prod mis-match
- **Deploy Audit –** verify deploy success, integrate with DevOps

Future Windows, Linux, Unix agents for multi-platform apps