adaptTo()

EUROPE'S LEADING AEM DEVELOPER CONFERENCE

28th – 30th SEPTEMBER 2020

Scanning for Malware in Apache Sling and Adobe Experience Manager

Oliver Lietz

Preface

Malware – malicious software

Keyloggers

Spyware

Trojan horses

Viruses

Adware

Rootkits

Ransomware

Potentially Unwanted Programs/Applications (PUP/PUA)

Backdoors

Worms

EICAR

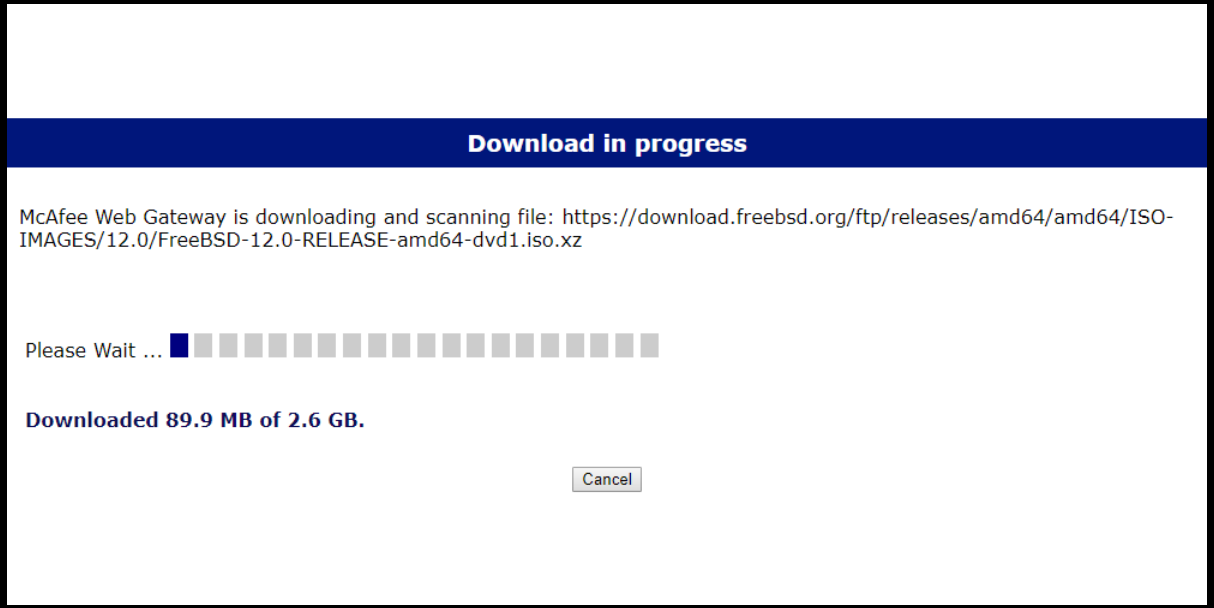
- European Institute for Computer Anti-Virus Research
- Expanded into other fields of IT security
- Distributes “The Anti-Malware Testfile” aka EICAR Standard Anti-Virus Test File

```
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

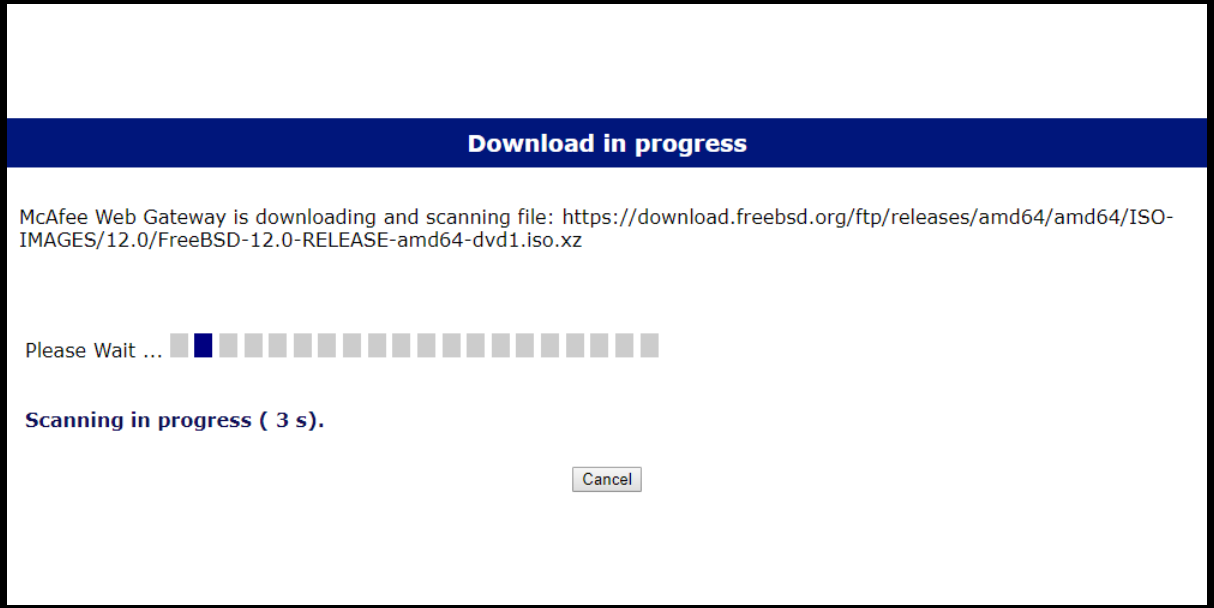
ICAP

- Internet Content Adaptation Protocol
- IETF RFC 3507
- Lightweight HTTP-like protocol
- Adapting HTTP requests and responses
- Generally used to implement virus scanning and content filtering in HTTP proxies

ICAP – Downloading

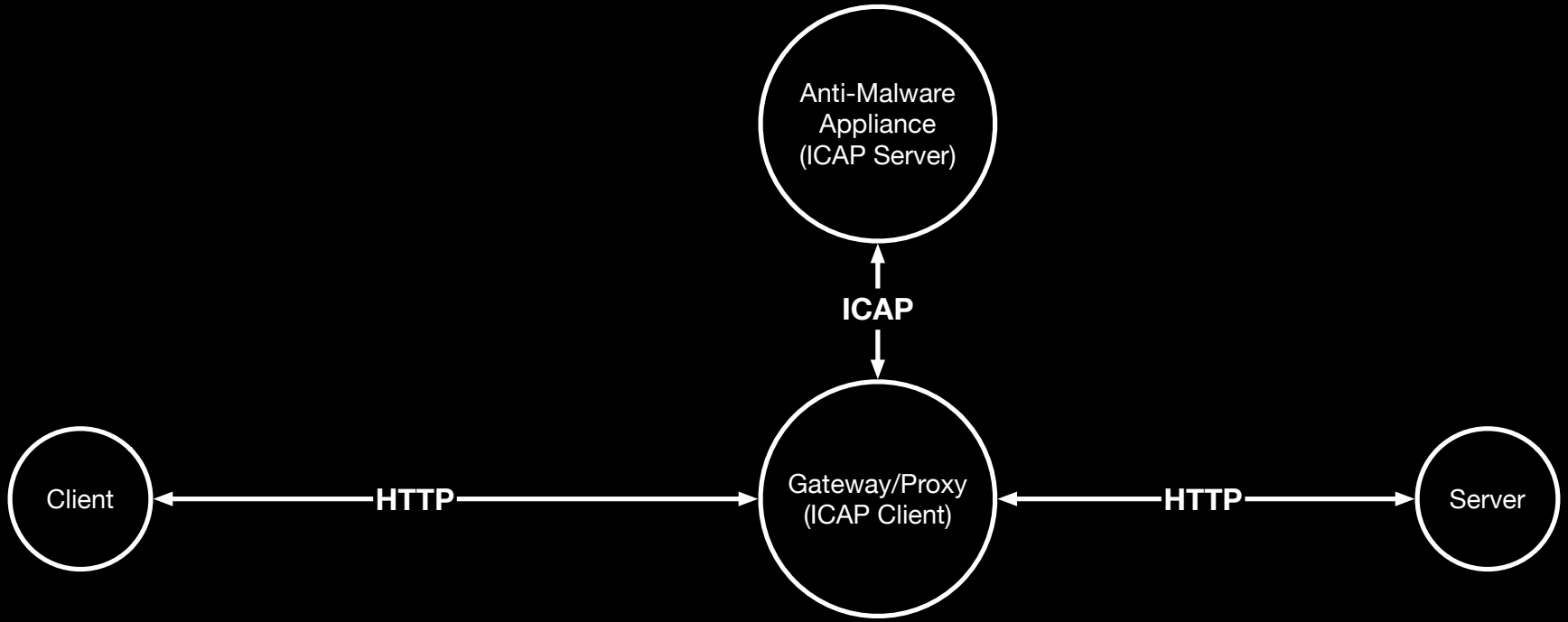


ICAP – Scanning



ICAP – Blocking

ICAP – Virus Scanning

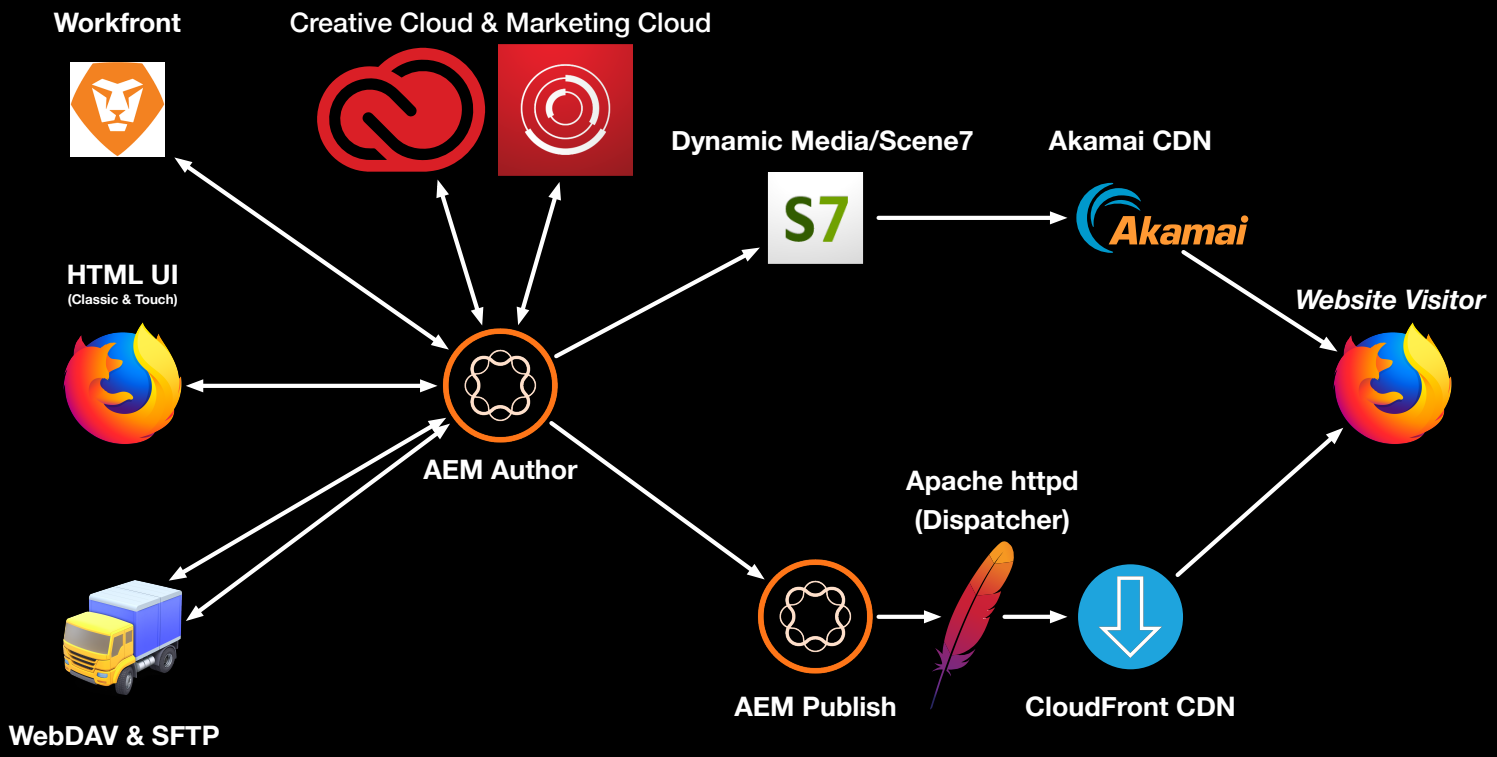


Use case and key points

- Prevent distribution of malware
  (loss of trust and revenue)
- Enterprise Content Management System
  (content shared globally and content per country)
- AEM 6.4 Sites and Assets
- Operated by Adobe Managed Services
- Hosted on Amazon AWS
- Dynamic Media/Scene7 for delivering assets
- Content managed by several agencies worldwide

adaptTo()

System landscape



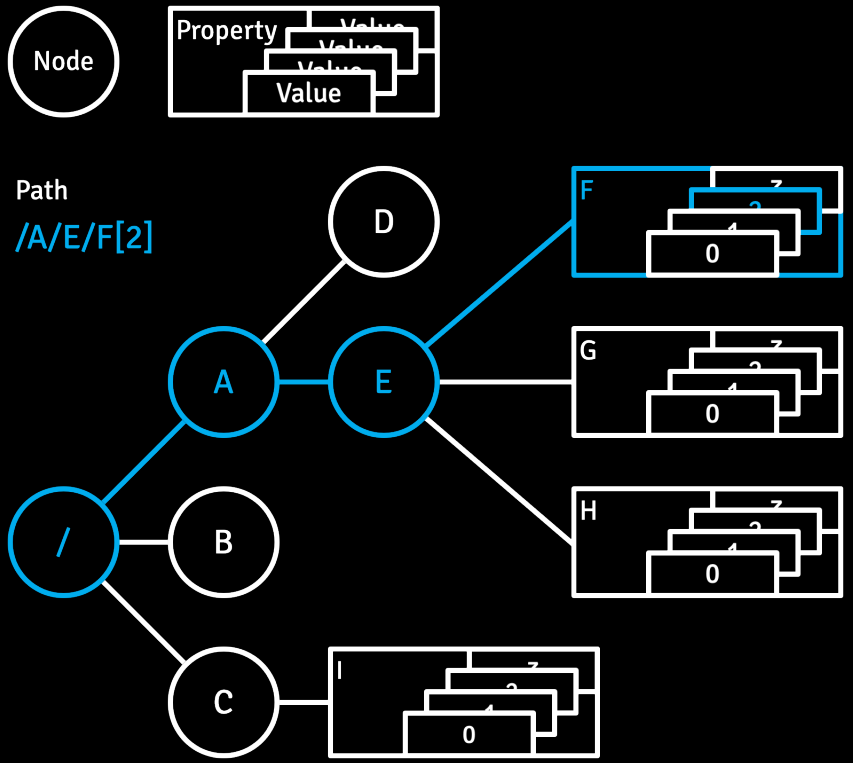
Uploading content into Sling and AEM

- Assets, packages, bundles
- Author UI and Web Console
- RESTful APIs
- WebDAV
- Creative Cloud, Marketing Cloud
- SFTP
- 3rd party integrations/connectors

Challenges

- Intercepting/blocking uploads (proxy)
  - Secure connections (HTTPS/SFTP)
  - Large binaries (high resolution raster images, videos)
  - Mixed content – binaries and plain text (meta data)
- Additional load (CPU/Memory) on AEM Author
- Prompt alerting

JCR – Logical model (node, property, path)



- Common node types

  `nt:folder`, `nt:file`, `nt:resource`, `nt:unstructured`, `dam:Asset`, `cq:Page`, `rep:User`

- Property types

  STRING, URI, BOOLEAN, LONG, DOUBLE, DECIMAL, **BINARY**, DATE, NAME, PATH, WEAKREFERENCE, REFERENCE

JCR – JSON rendering (asset example)

/content/dam/we-retail/en/products/activities/running/gomobile-source/Water Bottle.jpg


```
{
  "jcr:primaryType": "dam:Asset",
  "jcr:mixinTypes": ["mix:versionable"],
  "jcr:createdBy": "admin",
  "jcr:versionHistory": "921c7c08-f040-411d-8cb0-c1c0382725e0",
  "jcr:predecessors": ["419871af-6463-47d5-aabb-4066f6c4bb62"],
  "jcr:created": "Wed Sep 23 2020 19:45:12 GMT+0200",
  "jcr:baseVersion": "419871af-6463-47d5-aabb-4066f6c4bb62",
  "jcr:isCheckedOut": true,
  "jcr:uuid": "69a166dc-3095-4f59-947e-d398f2079438",
  "jcr:content": {
    "jcr:primaryType": "dam:AssetContent",
    "dam:s7damWidth": 717,
    "dam:s7damHeight": 1080,
    "jcr:lastModifiedBy": "admin",
    "dam:s7damAnchorX": 358,
    "dam:s7damAnchorY": 540,
    "dam:lastInvalidateCache": "Sat Jan 16 2016 23:34:56 GMT-0500",
    "dam:s7damType": "Image",
    "cq:name": "Water Bottle.jpg",
    "dam:assetID": "69a166dc-3095-4f59-947e-d398f2079438",
    "jcr:lastModified": "Mon Sep 12 2016 17:17:01 GMT+0530",
    "cq:parentPath": "/content/dam/we-retail/en/products/activities/running/gomobile-source",
    "dam:relativePath": "we-retail/en/products/activities/running/gomobile-source/Water Bottle.jpg",
    "renditions": {
      "jcr:primaryType": "nt:folder",
      "jcr:createdBy": "admin",
      "jcr:created": "Wed Sep 23 2020 19:45:12 GMT+0200",
      "cq5dam.thumbnail.48.48.png": {},
      "cq5dam.thumbnail.140.100.png": {},
      "cq5dam.web.1280.1280.jpeg": {},
      "original": {
        "jcr:primaryType": "nt:file",
        "jcr:createdBy": "admin",
        "jcr:created": "Wed Sep 23 2020 19:45:12 GMT+0200",
        "jcr:content": {
          "jcr:primaryType": "nt:resource",
          "jcr:lastModifiedBy": "admin",
          "jcr:mimeType": "image/jpeg",
          "jcr:lastModified": "Wed Sep 23 2020 19:45:12 GMT+0200",
          "jcr:data": 43861,
          "jcr:uuid": "16019fc3-0298-433e-b457-de2dd803fa41"
        }
      },
      "cq5dam.thumbnail.319.319.png": {}
    },
    "related": {"jcr:primaryType": "nt:unstructured"},
    "metadata": {}
  }
}
```


JCR – Physical model (Oak)

- Node Storage: Oak Segment Tar
  - Tar files on file system of virtual machine
  - Default maximum segment size is 256MB
  - Amazon AWS EBS (Elastic Block Store)
- Blob Storage: (Shared) S3 Data Store
  - Chunked binaries
  - Default minimum size of a binary stored in the data store is 16KB, smaller binaries are inlined in the node store
  - Shared (between Author and Publish instances)
  - Amazon AWS S3 (Simple Storage Service)

adaptTo()

McAfee VirusScan Command Line Scanner

- Installed on AMS Linux instances
- Weekly updates and scans on the Author and Publish instances
- Update of signatures happens shortly before scan
- McAfee does not recommend invoking CLS for single file scanning on a repeated basis
- Executable: `/usr/local/av/uvscan`

McAfee VirusScan Command Line Scanner

/usr/local/av/scanfiles.txt

```
/usr
/var
/opt
/home
/tmp
/etc
/mnt
/lib
/lib64
/root
/misc
/sbin
```

McAfee VirusScan Command Line Scanner

```
/var/log/av/latestscan.log

McAfee VirusScan Command Line for Linux64 Version: 6.0.3.356
Copyright (C) 2010 McAfee, Inc.
(408) 988-3832 LICENSED COPY - September 09 2018

AV Engine version: 5700.7163 for Linux64.
Dat set version: 9402 created Oct 6 2019
Scanning for 668684 viruses, trojans and variants.

[...]

No file or directory found matching /misc


Summary Report on /usr/local/av/scanfiles.txt
File(s)
        Total files:.....       178726
        Clean:.....             178673
        Not Scanned:.....       53
        Possibly Infected:.....         0

Time: 01:39.58
```

McAfee MVISION Cloud for AWS

- Cloud Access Security Broker (CASB)
- Formerly Skyhigh Networks
- Includes malware detection and removal
- On-demand and real-time scanning
- Connects to S3 (Oak Blob Storage)

ClamAV

- Open source antivirus engine for detecting malware
- Supports multiple file formats and archive unpacking
- Includes a multi-threaded scanner daemon
- Includes command line utilities for on demand file scanning
- Automatic signature update
- Supports multiple signature languages

ClamAV Daemon - `clamd`

- Daemon listens for incoming connections on Unix and/or TCP socket
- Scans local files or directories on demand
- **`INSTREAM`** – Scans a stream of data

adaptTo()

filter detect report defend

Apache Sling Commons Clam

- Provides an interface to Clam daemon (OSGi Service calling `INSTREAM` on `clamd`)
- Runs in any OSGi (DS/SCR) container
- No dependencies on Sling or JCR

adaptTo()

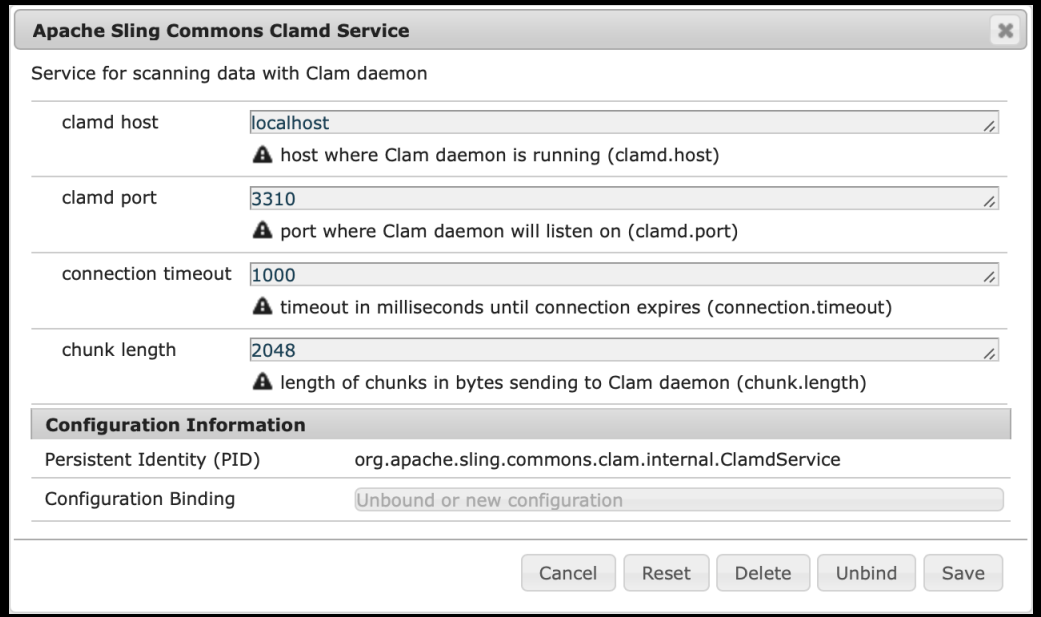
Apache Sling Commons Clam - API

```
public interface ClamService {

    ScanResult scan(InputStream data) throws IOException;

}
```

Apache Sling Commons Clam - API

```
public class ScanResult {

    [...]

    public long getTimestamp() {[...]} // ScanResult created

    public Status getStatus() {[...]} // Status from clamd

    public String getMessage() {[...]} // Message from clamd

    public long getStarted() {[...]} // Streaming to clamd started

    public long getSize() {[...]} // Number of bytes streamed to clamd

    public boolean isOk() {[...]} // Status == OK

    public enum Status {
        OK,
        FOUND,
        ERROR,
        UNKNOWN
    }

}
```

Apache Sling Commons Clam - Clamd Service



adaptTo()

Apache Sling Clam

- Hooks into JCR/Oak
- Service which observes the repository for writes (add, change)
- Service for on-demand digging
- `JobConsumer` to call Clam service for scanning
- `ScanResultHandlers` to report and alert
- HTTP API for on-demand digging and event listening

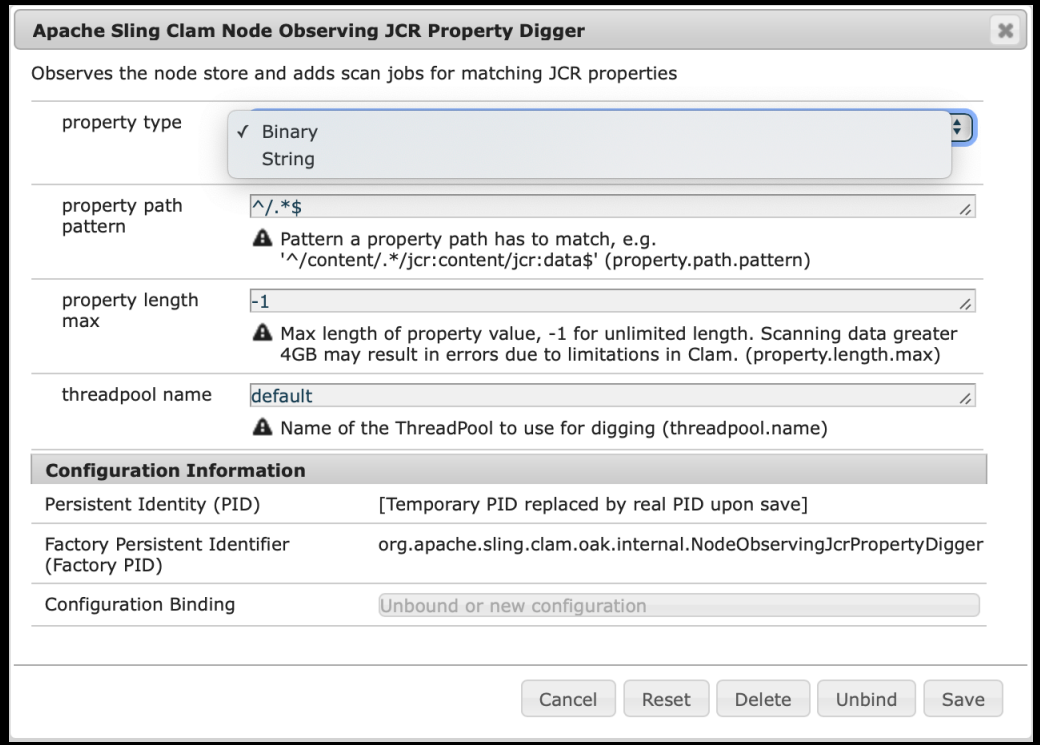
Apache Sling Clam - API

```
public interface NodeDescendingJcrPropertyDigger {

    void dig(
        Node node, // The entry node for digging
        Pattern pattern, // The pattern a property path has to match
        Set<Integer> propertyTypes, // The property types to take into account
        long maxLength, // The maximum length of a property value
        int maxDepth // The maximum depth from entry node for digging
    ) throws Exception; //

}
```

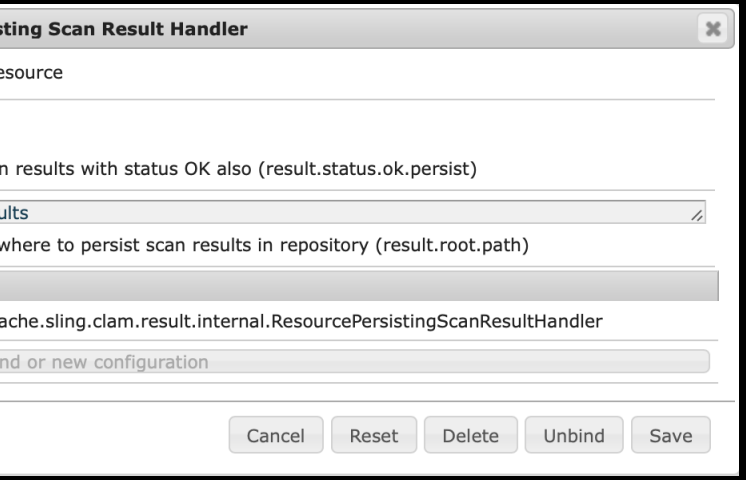
Apache Sling Clam - API

```
public interface JcrPropertyScanResultHandler {

    void handleJcrPropertyScanResult( // handles single-value property scan result
        ScanResult scanResult, // The scan result from Clam service
        String path, // The path of the scanned single-value property
        int propertyType, // The type of the scanned property
        String userId // The id of the user who added or changed the property
    );

    void handleJcrPropertyScanResult( // handles multi-value property scan result
        ScanResult scanResult, // The scan result from Clam service
        String path, // The path of the scanned multi-value property
        int index, // The index of the scanned property value
        int propertyType, // The type of the scanned property
        String userId // The id of the user who added or changed the property
    );

}
```

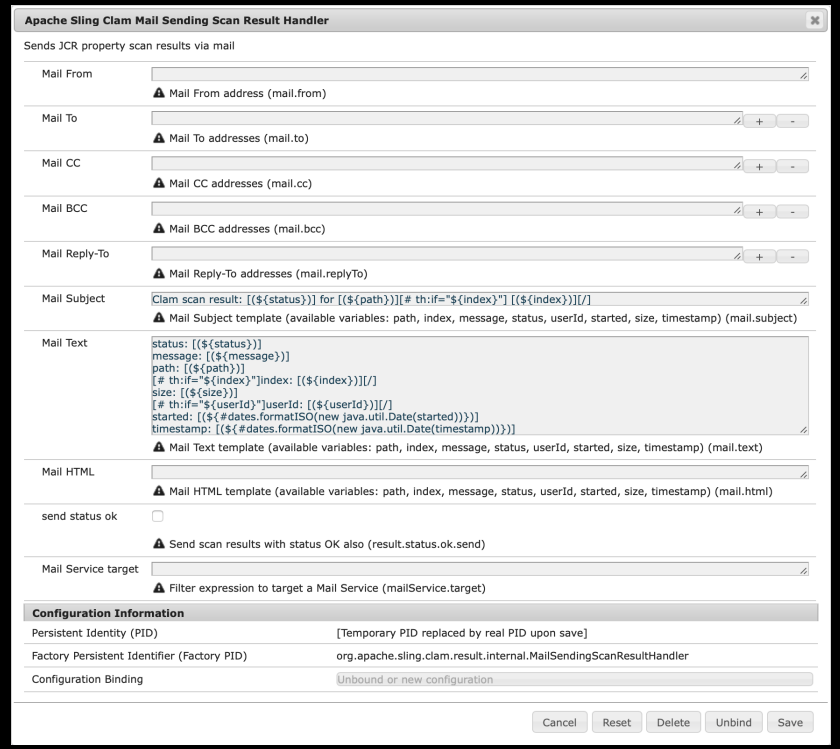
Apache Sling Clam – Node Observing JCR Property Digger



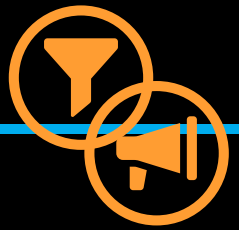
Apache Sling Clam – Resource Persisting Scan Result Handler



Apache Sling Clam – Mail Sending Scan Result Handler



Apache Sling Clam – HTTP API



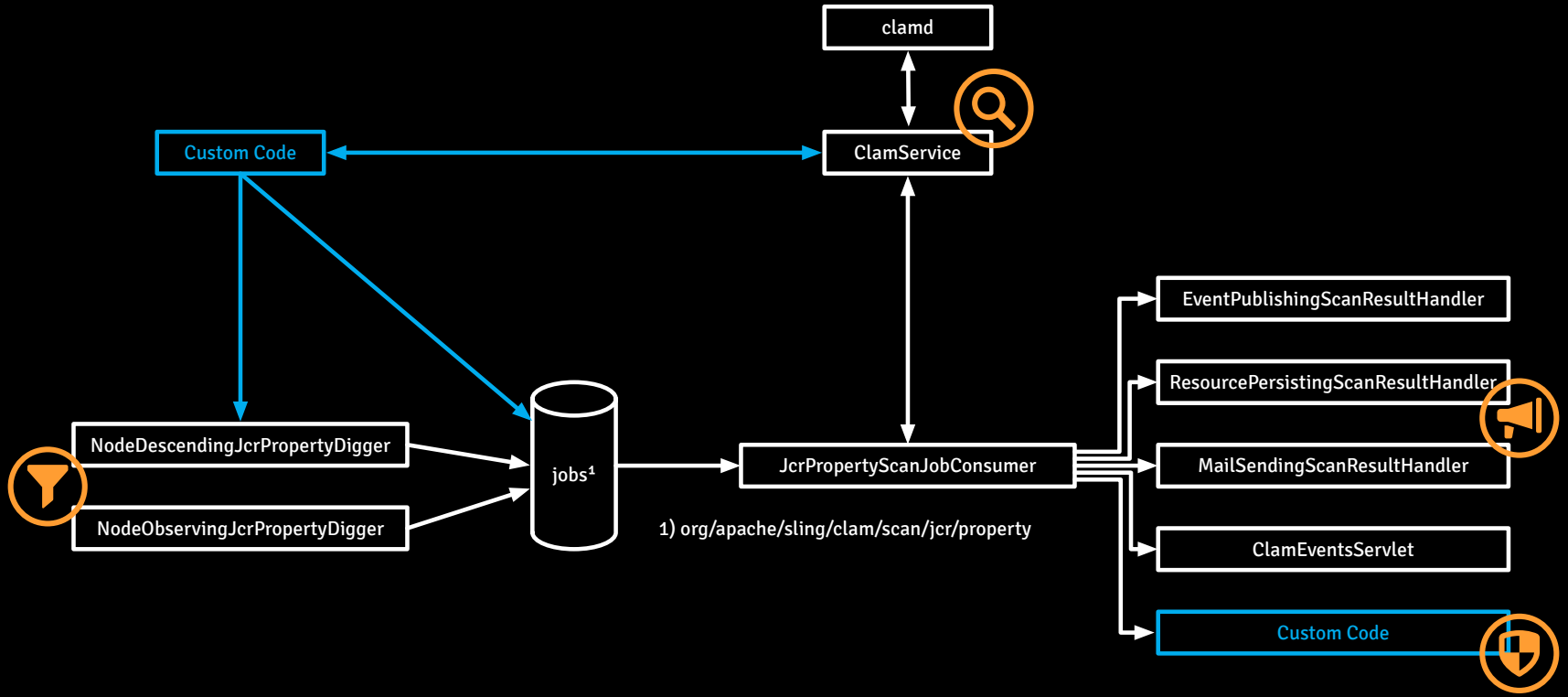
Scanning all binaries in AEM Assets:

```
curl -v -u username:password -F path=/content/dam\
http://localhost:4502/system/clam-jcr-scan
```

Listening to Sling Clam events:

```
curl -v -u username:password\
http://localhost:4502/system/clam-events
```

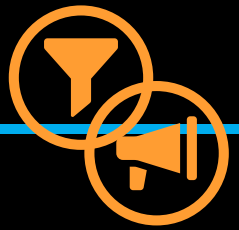
Apache Sling (Commons) Clam – Big picture



How to prevent malware distribution?

- Highly depends on business processes and technical integrations
- Use `JcrPropertyScanResultHandler` to…
  - Stop replication completely
  - Manage deny list and/or allow list for use with e.g. `ReplicationContentFilter`
  - Remove read access from data
  - Remove data from repo
  - …

Apache Sling Clam - Summary



- Agnostic to storage
  (Tar, S3, MongoDB, ...) acts on JCR/Oak API level
- Knows where content is stored in JCR (for removal)
- Non-blocking
  no impact on clients (humans and integrations/connectors)
- Low additional load on Author
- No leaking of content into additional 3rd party systems

adaptTo()

Commons Content Analyzing/Processing API

- New APIs for Content Analyzing and Content Processing
- Analyzing: content type detection, malware detection, ...
- Processing: image compression, malware removal, ...

```
public interface ContentAnalyzer/ContentProcessor {

    CompletableFuture<Void> analyze/process(
        InputStream input, // The stream from which the content is read for analyzing/processing
        OutputStream output, // The stream into which the content is written during or after processing
        Map<String, Object> parameters, // The parameters for the analyzing/processing operation
        Map<String, Object> report // The report to which the findings of the analyzing/processing operation are added
    );

}
```

adaptTo()

What's next?

- Making clamd service a content analyzer (WIP)
- Making Sling Clam a general Sling AntiVirus
- Providing a HTML UI (Angular or similar)

What else?

- Sanesecurity: Provides additional malware signatures for ClamAV
- SecuriteInfo: Provides additional malware signatures for ClamAV (commercial)
- VirusTotal: Portal providing public API to scan samples and search for malware
- mk0x/docker-clamav: Docker Image with Clam Daemon

adaptTo()

Summary

- Processing clean content is the normal case, infected content is the exceptional case
- Be prepared for the exceptional case
  - Have processes in organization established
  - Practice, don’t panic
- Never (re)move infected files via file system, always use JCR API – otherwise you will break your repository
- Prevent distribution of infected content

Resources

- Apache Sling Commons Clam
  https://github.com/apache/sling-org-apache-sling-commons-clam
- Apache Sling Clam
  https://github.com/apache/sling-org-apache-sling-clam
- Bildschirmarbeiter AEM Clam Package
  https://github.com/bildschirmarbeiter/de.bildschirmarbeiter.aem.packages.clam
- SfMiASaAEM Package
  https://github.com/oliverlietz/sfmiasaaem-package
- users@sling.apache.org