Integrated Visualization of Network Security Metadata from Heterogeneous Data Sources

Bastian Hellmann

Trust@HsH Research Group
University of Applied Sciences and Arts in Hanover

July 13th 2015
GraMSec 2015
Verona, Italy

HOCHSCHULE
HANNOVER
UNIVERSITY OF
APPLIED SCIENCES
AND ARTS
Fakultät IV
Wirtschaft und
Informatik

Motivation

Problem in Network Security

- Typical networks consist of various components like user endpoints, network security devices, services, ...
- Information is not shared among those components.
- Thus, an overview of *whats going on* is difficult.

Exemplary Use-Cases

- Detect combinations of failed login attempts on multiple services by the same user.
- Find the sources of the attack.
- Trace the way the attack *moved* in the network.
- React fast by shutting down accounts or locking out devices.

Our Contribution

- Design and implement an integrated visualization, that works with data from various sources, and helps to detect and react to such attacks.

Requirements

Real-Time Monitoring

- Acquire knowledge before it is outdated
- → Allows for faster reactions after detection of abnormal behavior

Data Integration

- Combine information and knowledge from arbitrary components
- → Allows to combine knowledge

Retrospective Analysis

- Preserve historical course of data and provide means to navigate in time
- → Events that led to a specific state can be retracted

Integration of Data Sources

Types of data interesting for network security

- Physical and logical topology
- Configuration of devices & services
- State
- Behavior

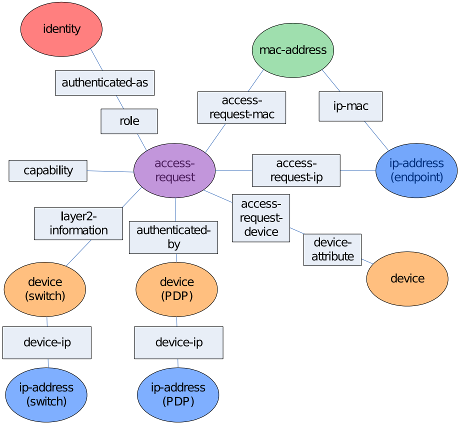
Our approach

- Use IF-MAP protocol as the foundation (→ next slides)

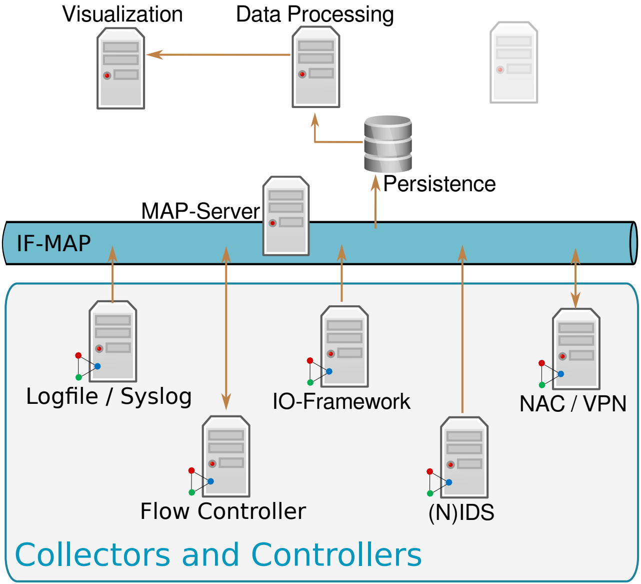
Interface for Metadata Access Points (IF-MAP)

What is it?

- Open specification by the Trusted Computing Group
- Goal: allow exchanging information between arbitrary network components
- Data is defined in an extensible way and not bound to a domain



Proposed Architecture



Application and Persistence Level Concepts

Enhancements to IF-MAP

- Persistence of IF-MAP data, as the MAP server only holds the current state

Continuous recording

- Preserve the *changes* (not only snapshots) as the MAP server receives them

State and change queries

- Allow to query for *snapshots*, i.e. the complete graph at a given time
- Allow to query for the *changes* (delta) between two timestamps

Visualization Requirements and Concepts I

Representing the data model

- IF-MAP forms a graph with nodes (identifier) and edges (links) and information attached to them
- Thus can be rendered with standard graph rendering techniques and layouts

Publisher distinction

- The source (i.e. the MAP client *measuring* the data) of metadata needs to be transparent to the user
- Use the IF-MAP *publisher-id* to distinct between metadata from different clients (e.g. by coloring)

Visualization Requirements and Concepts II

History navigation

- Navigation via three modes: *live view*, *history view*, *delta view*
- Selection of timestamps via *slider* and/or forward-backward-buttons

Search Functionality and Filtering

- Allow the user to search or filter the graph data, to pinpoint a single node or a selection of nodes with similar features
- Search results can either be highlighted or colored differently as non-matching nodes
- Non-matching nodes also can be shown translucent, to retain the overall structure

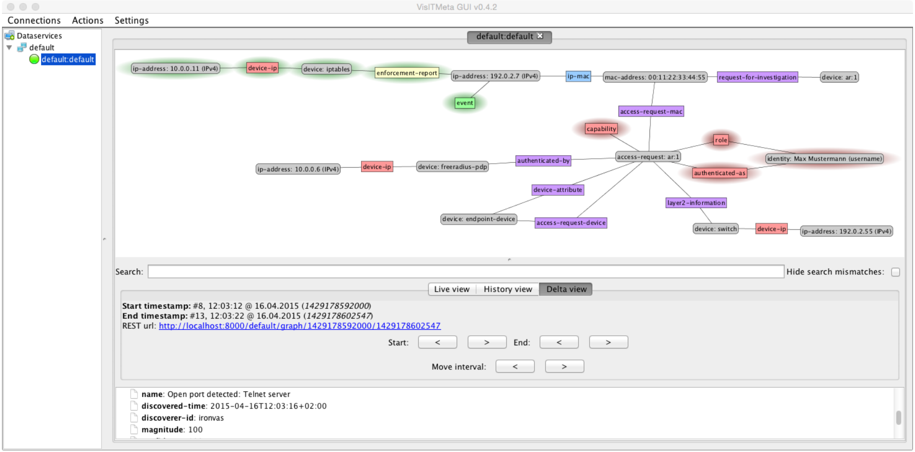
VisITMeta Application

General information

- Research project, funded by German Ministry for Research and Education
- Released as open-source software[1]
- Implementation of all previous concepts
- Offers additional features like motion control (LeapMotion)

[1] https://github.com/trustathsh/visitmeta

Screenshot (v0.4.2)



Search:

Hide search mismatches: □

Start timestamp: #8, 12:03:12 @ 16.04.2015 (*1429178592000*)
End timestamp: #13, 12:03:22 @ 16.04.2015 (*1429178602547*)
REST url: http://localhost:8000/default/graph/1429178592000/1429178602547

name: Open port detected: Telnet server
discovered-time: 2015-04-16T12:03:16+02:00
discoverer-id: ironvas
magnitude: 100

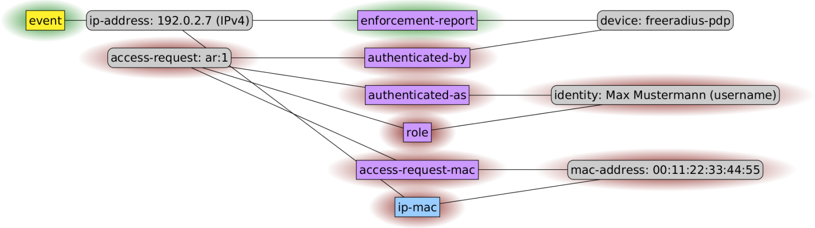
Specifics of IF-MAP Visualization

Characteristics of IF-MAP graphs

- Two kinds of nodes: Identifier and Metadata
- Can and should be handled differently when calculating layouts

Example: Adapted Bipartite Layout

- Identifier in columns 2 and 4, Metadata in columns 1, 3 and 5
- Emphasizes the difference between link and single metadata



Results

Homogenization

- IF-MAP used for acquisition and homogenization of different data sources
- Components need only be enabled to publish IF-MAP information

Data context

- Implicit connection of different data like network addresses, user credentials, services and high-level events

Interoperability

- VisITMeta as a software is usable in every IF-MAP-based environment as it uses standard mechanisms to fetch the data
- Many MAP clients and thus a good amount of data sources already available

Continuous recording and retrospective analysis

- Changes are persisted as they are processed by the MAP server
- They can then be reconstructed step by step

Indentified Challenges

Visual Scalability

- Big graphs get cluttered really quick
- Techniques to reduce the size of the graph have to be added, like Level of Detail

Visual dynamics

- Frequent changes in the network lead to many changes in the visualization
- E.g. do not show low-level data and concentrate instead on high-level abstractions

Recording of all data

- Mechanism to fetch data from the MAP server has a shortcoming implied from IF-MAP itself: only connected graphs can be observed via a *subscription*
- IF-MAP does not offer a mechanism to get to know if there are new and disconnected graphs.

Conclusion

Summary

- Requirements for data integration including model and requirements for visualizing the data
- IF-MAP as foundation
- Graph-based visualization with regards to IF-MAP structure
- Features like history navigation and filtering

Future Work

- Address the identified challenges
- Using data persistence for analysis