Visualization of Network Security Policy Evaluation

Bastian Hellmann[*] Marcel Reichenbach[†] Leonard Renners[‡] Volker Ahlers[§]

University of Applied Sciences and Arts Hannover, Germany
Faculty IV, Department of Computer Science
Home page (Trust@HsH research group): http://trust.f4.hs-hannover.de/

Abstract

Network security components often store and process their information in proprietary ways. Doing analysis on data of different components is therefore limited or requires a prior homogenization step. Visualizing this data is also often limited to suitable visualization schemes for single components.

The Interface for Metadata Access Points (IF-MAP) specification by the Trusted Computing Group helps to combine the data of multiple different components in a homogenized graph-based data format. This allows to use the data for analysis that takes into account all gathered information and thus to get a more educated view on the state of a network. It also helps visualizing the data for the user, as relationships between the data can be presented clearly.

But even with IF-MAP that provides homogenization and centralized collection of data, another type of information is still missing: Analysis components that use data to detect unwanted behavior often do not share their inner state and processing, so that understanding them in cases of wrong or unexpected results is not easy. A user then has to collect all information that lead to the specific result manually, by combining information from different sources. An approach to help the user in this task could be to publish the inner state and configuration of analysis components, ideally in the same data format as the sensor data, and thus allowing to see connections between measured data and the parts of a policy of an analysis component that lead to a specific evaluation result.

Our contribution shown in this abstract and the corresponding poster describes the IF-MAP specification and its benefits for data homogenization. A suitable visualization of this data by means of graph drawing is shown as the result of a research project called VisITMeta. We then propose an approach to visualize the configuration, i.e. policies, of analysis components within the same view in order to create relationships between sensor data and policy elements belonging to a successfully triggered policy evaluation.

1 Introduction

The Interface for Metadata Access Points (IF-MAP) specification [4] by the Trusted Computing Group defines an interoperable data and communication model to exchange metadata in a network. The data within IF-MAP is structured as a unweighted and undirected graph, with IF-MAP data types corresponding to nodes and edges. The data model itself can be applied to almost any domain, e.g. network security [3]. By specifying a publish/subscribe system with a central Metadata Access Point Server (MAPS) and arbitrary MAP Clients, data that was hitherto available only to the system that recorded it can now be aggregated and combined with data from other sources.

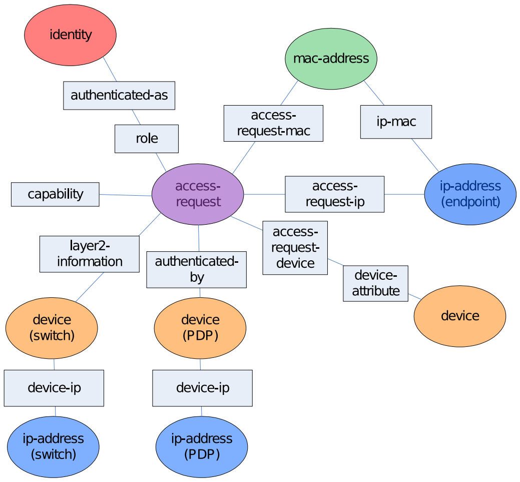

Figure 1: Example of a MAP graph.

The data model defines *Identifiers* that represent unique entities like an IP address or a device. Identifiers can be connected to each other by *Links*. Both Identifiers and Links can be enhanced with further information, called *Metadata*.

Figure 1 shows an example MAP graph with Identifiers presented as circles and Metadata as rectangles. The underlying information can be gathered by different sensors like a DHCP server managing ip-mac leases, a policy decision point handling user data and a service discoverer component like nmap measuring device characteristics. As information is published by the components, the relations to other parts of the graph are created implicitly, without the components knowing each other.

2 VisITMeta Project

VisITMeta is a recently finished research project focusing on the development of an open-source tool for the visualization of IF-MAP metadata graphs while being interoperable with other IF-MAP compliant clients and servers, both from previous projects and third-party vendors. In the current version of the software the following main features are implemented [2]:

Persistence of history and interface. Store and provide the current state and history of the graph, and develop an interface for arbitrary further processing of that history data.

Calculation and visualization of graph deltas. As the changes of the state within a network are interesting for analyzing incidents, the difference between two states are calculated and shown as graphs representing all updates and deletions.

Different layouts. Different layout algorithms can be used with the software; some algorithms like a standard Force-directed layout as well as standard algorithms adapted to the specifics of IF-MAP, e.g. a bipartite layout with columns for Identifiers and Metadata.

Filters and searches. Allow to filter specific metadata and allow to search for metadata by value or type.

[*] e-mail:bastian.hellmann@hs-hannover.de
[†] e-mail:marcel.reichenbach@stud.hs-hannover.de
[‡] e-mail:leonard.renners@hs-hannover.de
[§] e-mail: volker.ahlers@hs-hannover.de

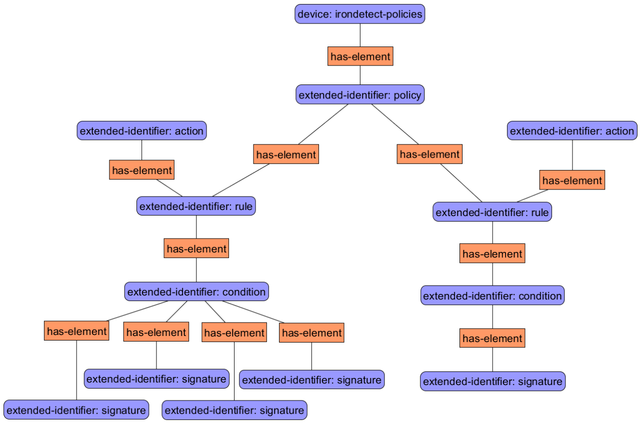
device: irondetect-policies
has-element
extended-identifier: policy
extended-identifier: action
has-element
has-element
extended-identifier: action
has-element
has-element
extended-identifier: rule
extended-identifier: rule
has-element
has-element
extended-identifier: condition
extended-identifier: condition
has-element
has-element
has-element
has-element
has-element
extended-identifier: signature
extended-identifier: signature
extended-identifier: signature
extended-identifier: signature
extended-identifier: signature

Figure 2: Sample visualization of a policy as IF-MAP graph.

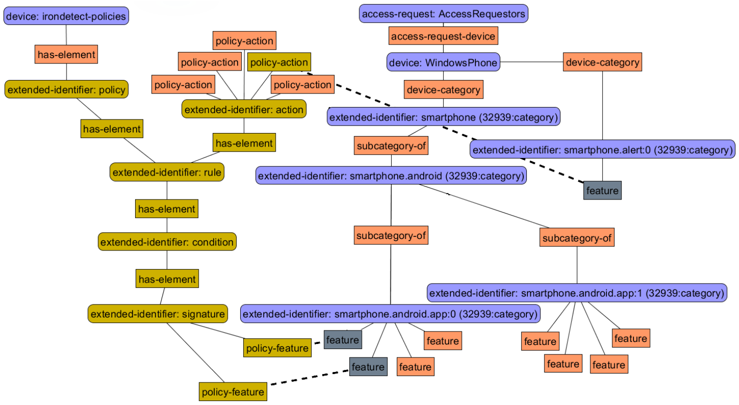
device: irondetect-policies
access-request: AccessRequestors
policy-action
access-request-device
has-element
policy-action
policy-action
device: WindowsPhone
device-category
extended-identifier: policy
policy-action
policy-action
device-category
extended-identifier: action
extended-identifier: smartphone (32939:category)
has-element
has-element
subcategory-of
extended-identifier: smartphone.alert:0 (32939:category)
extended-identifier: rule
extended-identifier: smartphone.android (32939:category)
feature
has-element
subcategory-of
subcategory-of
extended-identifier: condition
has-element
extended-identifier: smartphone.android.app:1 (32939:category)
extended-identifier: signature
extended-identifier: smartphone.android.app:0 (32939:category)
feature
feature
feature
feature
policy-feature
feature
feature
feature
feature
policy-feature

Figure 3: Sample visualization of a policy evaluation result.

Highlighting changes. Use animations and glow-effects to highlight changes in the graph.

Interchangeability. Easily exchange libraries and algorithms for persistence, layout, rendering, etc.

The architecture consists of two separated applications: the *dataservice* and the *visualization* component. The dataservice collects the metadata from a MAP server via IF-MAP and stores it inside a graph database. It also provides access to the stored metadata via a REST-like interface. The connection to the MAP server is done via the IF-MAP client library ifmapj, while a neo4j[1] database is used for persistence [1].

The visualization application fetches the metadata from the dataservice via the REST-interface and converts the data into graph elements. Layouts are generated with the JUNG2[2] library and the result are rendered with Piccolo2D[3]. A simple GUI allows for navigating through the metadata, selecting the point of time (or even two points in time to get a delta view of the metadata) [2].

3 Visualizing Policy Evaluation Results

As an enhancement to showing the sensor data, representing the state and the events happening in the network, insight to analysis processes that result in events in the data is helpful. Therefore both the configuration, i.e. the policy of a component, and evaluation results with their dependencies to involved sensor data need to be mapped onto IF-MAP. This is discussed below for a component called *irondetect*, which evaluates IF-MAP data using rules that combine conditions consisting of of signatures and anomalies with actions to be performed when a rule is fulfilled.

As a first step, the policy model of irondetect was mapped onto IF-MAP data entities. Figure 2 shows the representation of an irondetect policy as an IF-MAP graph.

After the policy itself is mapped and visualized, concrete evaluation results can be shown. Figure 3 shows a graph that depicts the evaluation of a specific rule and the connection to the corresponding sensor data. It also connects the action that was defined for the rule and its result, i.e. a new metadatum in the graph. To emphasize this connection as an especially important one, it is drawn in another style as the common IF-MAP link connections. Thus the user can directly see and then analyse both the policy elements and their values that lead to an incident as well as the corresponding IF-MAP data that lead to its successful evaluation. This could help in cases of false positives to narrow down the misconfiguration without manually combining the needed information from different data sources and representations.

4 Current Progress and Future Work

The VisITMeta software itself as well as irondetect are already available as open source projects on Github[4], together with a couple of other IF-MAP components usable to create an IF-MAP based network. Our proposed enhancements to visualize policies and evaluation results are not released yet but will become public in the next few months.

In the future we would like to study how the user can react to evaluation results presented within the GUI and modify elements of the policies in order to manage misconfigured components in the case of false positives. We also want to adapt further analysis components so that they share their inner state and configuration in IF-MAP manner, especially components with different policy schemes such as policies of graph-pattern matching components.

Acknowledgements

The VisITMeta project was funded by the German Ministry for Research and Education (BMBF, grant no. 17PNT032).

References

[1] V. Ahlers, F. Heine, B. Hellmann, C. Kleiner, L. Renners, T. Rossow, and R. Steuerwald. Replicable security monitoring: Visualizing time-variant graphs of network metadata. In J. Burton, G. Stapleton, and K. Klein, editors, *Joint Proceedings of the Fourth International Workshop on Euler Diagrams (ED 2014) and the First International Workshop on Graph Visualization in Practice (GViP 2014) co-located with Diagrams 2014*, number 1244 in CEUR Workshop Proceedings, pages 32–41. CEUR-WS.org, 2014.

[2] V. Ahlers, F. Heine, B. Hellmann, C. Kleiner, L. Renners, T. Rossow, and R. Steuerwald. Integrated visualization of network security metadata from heterogeneous data sources. In S. Mauw, B. Kordy, and S. Jajodia, editors, *The 2nd International Workshop on Graphical Models for Security, GraMSec 2015*, Lecture Notes in Computer Science. Springer, 2015. To appear.

[3] Trusted Network Connect Working Group. TNC IF-MAP metadata for network security, version 1.1, revision 8, May 2012. `http://www.trustedcomputinggroup.org/resources/tnc\_ifmap\_metadata\_for\_network\_security`.

[4] Trusted Network Connect Working Group. TNC IF-MAP binding for SOAP, version 2.2, revision 9, March 2014. `http://www.trustedcomputinggroup.org/resources/tnc_ifmap_binding_for_soap_specification`.

[1] `http://www.neo4j.org/`

[2] `http://jung.sourceforge.net/`

[3] `http://www.piccolo2d.org/`

[4] `https://github.com/trustathsh/`