# Security Events RISC Use Cases

Marius Scurtescu, Google
IETF99 Prague
July 2017

# Overview

- RISC use cases for security events
- provides requirements for token format and event distribution

# Definitions

- Transmitter - the entity that sends security events
- Receiver - the entity that receives security events
- IdP - Identity Provider, in most cases but not always this is the transmitter
- RP - Relying Party, in most cases but not always this is the receiver
- RISC - Risk and Incident Sharing and Coordination
- SCIM - System for Cross-domain Identity Management
- Explicit IdP to RP relationship - The RP is registered with the IdP and users go through an OAuth 2 flow.
- Implicit IdP to RP relationship - The RP is using a personal identifier controlled by the IdP, without formal registration. For example, the RP uses email addresses and the IdP is the email provider.

# Case 1: Explicit IdP to RP

- Transmitter: IdP
- Receiver: RP

Simplest use case, IdP sends security events to relevant RPs.

RP can make control plane calls to the IdP and can authenticate with access tokens issued by IdP.

# Case 2: Explicit RP to IdP

- Transmitter: RP
- Receiver: IdP

The RP can also send RISC events back to IdP. We want to make it very easy for the RP to do that, no complicated registration steps and crypto of possible.

IdP can document endpoint for data plane (where it receives events). RP can use access token when sending events on data plane and may not need to sign SETs.

RP could expose full transmitter functionality. In this case the IdP needs to register with the RP as an OAuth 2 client so it can use Access Tokens with control plane.

# Case 3: Implicit IdP to RP 1/2

- Transmitter: implicit IdP
- Receiver: implicit RP

IdP and RP need legal agreement.

When RP account is created/updated with IdP controlled personal identifier (i.e. email) then the RP makes an API call to the IdP to enroll this identifier for security events.

**Assumption**: RP is registered with IdP as an OAuth 2 client and can use Access Tokens with control plane.

# Case 3: Implicit IdP to RP 2/2

**Open questions**:

- What are the implications of unverified personal identifiers (i.e. unverified email addresses)?
- How does the RP find the correct IdP for a given identifier? For the email case this is the hosted domain discovery issue.

# Case 4: Implicit RP to IdP

- Transmitter: implicit RP
- Receiver: implicit IdP

No enrollment call is strictly necessary. The RP can start sending events to IdP as new identifiers show up.

**Assumption**: IdP is registered with RP as an OAuth 2 client and can use Access Tokens with control plane. This is the reverse of the normal registration.

**Open question**:

- Is enrollment needed?

# Case 5: Pseudo-implicit

Common personal identifier (i.e. email) used by two different RPs.

RPs need legal agreement.

Mutual discovery by exchanging identifier hashes.

**Open question**:

- What are the legal and privacy implications?

# Case 6: Identity as a Service

IdaaS should be able to manage SET distribution configuration for its RPs with a given IdP using the credentials already established between the RP and the IdP. Control plane operation to create/update stream allows that.

**Assumption**: IdaaS can impersonate RP at IdP (can obtain access token on behalf of RP).

# Case 7: Security as a Service

Similar to IdaaS described in previous section, but the service provider (SP) has its own set of credentials different from the credentials that an RP is using. The SP cannot impersonate the RP at IdP.

The IdP must define delegation rules and allow the SP to make requests on behalf of the RP.

# Case 8: On-Premise RP

- Transmitter: IdP
- Receiver: RP

The RP is behind a firewall and cannot be reached through HTTP.

The only way to deliver events is by periodical polls done by the receiver to an endpoint provided by the transmitter.

# Q & A