TURN Extension for Third Party Authorization

draft-ietf-tram-turn-third-party-authz-00

July 2014 IETF 90 Meeting

Authors : T.Reddy, P.Patil, J. Uberti, R.Ram

Changes from individual 00 to WG doc

- Moved from handle to self-contained OAuth token based on feedback from WG.

Handle Token verses Self-contained token

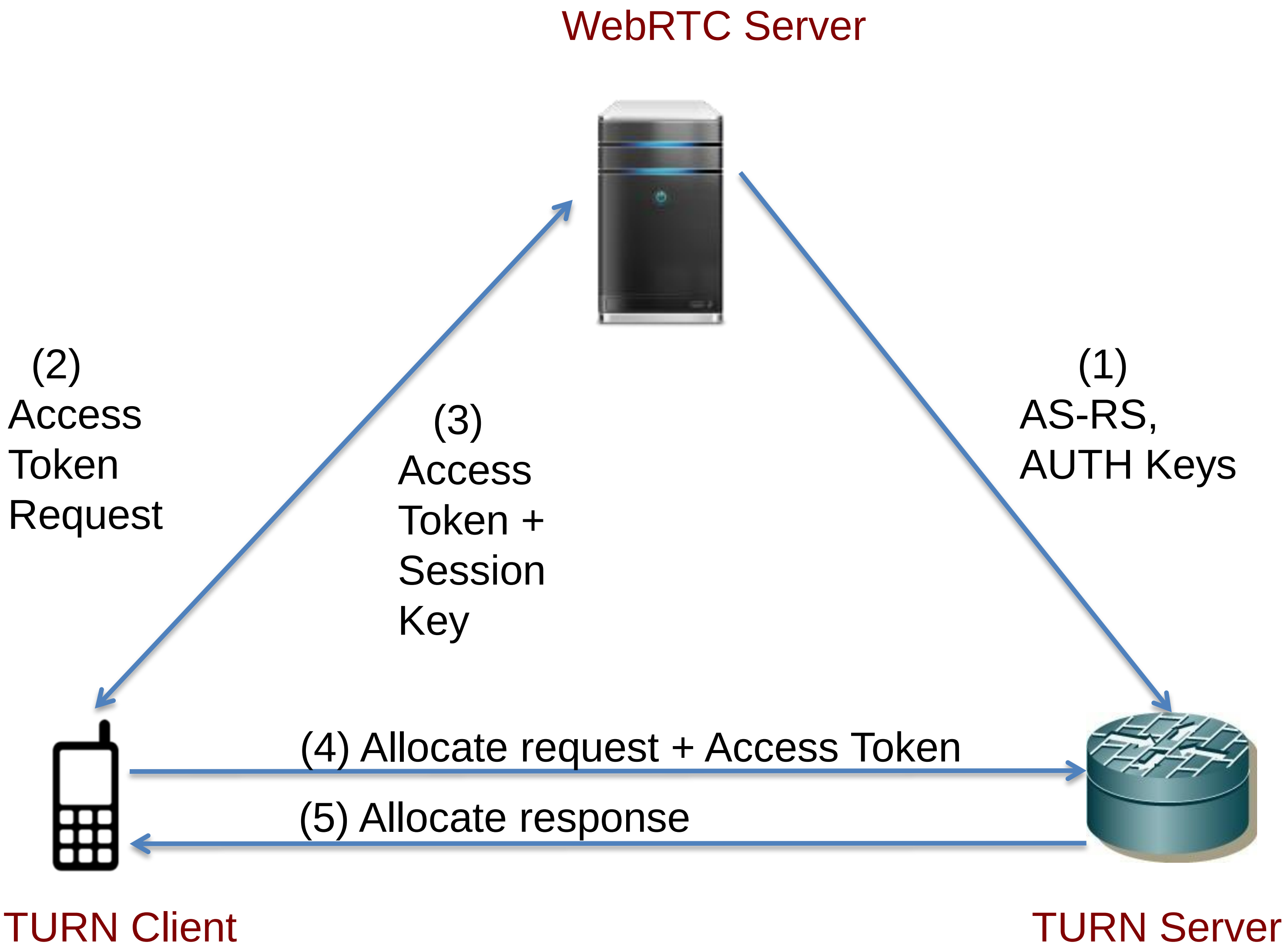
Self-Contained Token

- Token which would carry session key, lifetime, timestamp etc.
- For example Token size is 223 bytes when session key 256-bit, 128-bit (HMAC-SHA-256-128), AES_256_CBC are used.

Handle Token

- Requires communication between TURN server and Authorization server.

3rd Party authorization for TURN using OAuth



WebRTC Server

(1) AS-RS, AUTH Keys

(2) Access Token Request

(3) Access Token + Session Key

(4) Allocate request + Access Token

(5) Allocate response

TURN Client

TURN Server

STUN Attribute : THIRD-PARTY-AUTHORIZATION

- Used by TURN server to signal that it supports third party authorization.
- Provides the TURN server name, used by the authorization server for selecting the keying material.

STUN Attribute : ACCESS-TOKEN

```
struct {
        opaque {
              ushort key_length;
              opaque mac_key[key_length];
              opaque timestamp[8];
              long lifetime;
          } encrypted_block;
         opaque mac[mac_length];
    } token;
```

USERNAME

- Re-used to signal **kid** which allows the authorization server and resource server to select the appropriate keying material for encryption and decryption.

draft-ietf-tram-turn-third-party-authz-00

Questions ?

Backup

Problem with STUN Authentication

Problems are discussed in draft-ietf-tram-auth-problems-02

3rd Party authorization for TURN using OAuth

OAuth	TURN
Client	TURN Client
Resource Owner	Authorization Server (e.g.: WebRTC/SIP server)
Authorization server	Authorization Server
Resource Server	TURN Server

Key Establishment b/w TURN and Authorization servers

- REST API
  - OAuth 2.0 Proof-of-Possession (PoP) Security Architecture (draft-hunt-oauth-pop-architecture-02)
- Dynamic Symmetric Key Provisioning Protocol (RFC 6063)
- Manual Provisioning

Advantages

Client sends HTTP request to WebRTC server to get ephemeral token.

- No long-term credentials to keep secret; even if discovered, credential usefulness is limited
- Username contains no externally-identifying information and helps to provide privacy.
- Session Key is machine-generated, to prevent dictionary attacks