HTTPS Token Binding with TLS Terminating Reverse Proxies


Brian Campbell

IETF 99
Prague
July 2017
Problem Statement

- HTTPS application deployments often have TLS ‘terminated’ by a reverse proxy (TTRP) sitting in front of the actual application
- For applications in such deployments to take advantage of token binding, some information needs to be communicated from the TLS layer to the application (in the general case anyway)
- In the absence of a standard means of doing this, different implementations will do it differently
  - Terrible for interoperability
  - A boon to unneeded complexity
  - Improved opportunity to get things wrong
  - i.e. client certificate authentication
Confirmation Bias...

- I’ve been out spreading the good word of Token Binding
- Often the question of “what if my application isn't the piece that does TLS?” comes up
A Short History

- IETF 97 Seoul: 'consensus to work on the problem'
  - Two general approaches possible:
    - Expose Token Binding ID(s)
    - Expose EKM
- draft-campbell-tokbind-tls-term-00 exposes EKM to backend as header
- TTRP acronym coined by =JeffH
- Received some pushback on approach (primarily from implementers working with NGINX and Apache)
- IETF 98 Chicago: rushed & cut short in main session due to time
  - But held an open side meeting later in the week
    - That group clearly favored approach of exposing Token Binding IDs
- draft-campbell-tokbind-ttrp-00 exposes Token Binding IDs to backend as headers
Definitions HTTP headers that enable a TTRP and backend server to function together as a single logical server side deployment of HTTPS Token Binding

- TTRP validates the TokenBindingMessage from the Sec-Token-Binding header and removes it from dispatched request
- Provided-Token-Binding-ID header with base64url encoded provided TokenBindingID added to dispatched request
- Referred-Token-Binding-ID header with encoded referred TokenBindingID (if applicable) added to dispatched request
- Trust between the TTRP and backend server
- TTRP required to sanitize headers
- Original TokenBindingMessage not provided to backend
... and Running Code

mod_token_binding

A pluggable module implementation of Token Binding for the Apache HTTPd web server version 2.4.x.

Overview

This module implements the Token Binding protocol as defined in https://github.com/TokenBinding/Internet-Drafts on HTTPS connections setup to mod_ssl running in an Apache webserver.

It then sets environment variables with the results of that process so that other modules and applications running on top of it can use that to bind their tokens and cookies to the so-called Token Binding ID. The environment variables are:

- **Provided-Token-Binding-ID**
  The Provided Token Binding ID that the browser uses towards your Apache server conform draft-campbell-tokbind-trpr-00.

- **Referred-Token-Binding-ID**
  The Referred Token Binding ID (if any) that the User Agent used on the "leg" to a remote entity that you federate with conform draft-campbell-tokbind-trpr-00.

One could also pass these results to the backend in a header as with e.g.:

```
RequestHeader set Provided-Token-Binding-ID "R(Provided-Token-Binding-ID)e"
```

It works!

```
DATE_LOCAL = Wednesday, 12-Jul-2017 22:46:26 UTC
Provided-Token-Binding-ID = AgBBQMVoqXZ2cuOGGs6qunLJ2dVBMKlqoU8IFX1pFtBqN9ZMBYH10hQzZ0W4CYeFPCyFJXO3DMjAjbyPnMu5PDDA
Referred-Token-Binding-ID = (none)
Token-Binding-Context = AA0CrP_HeCYS-LWMuT0dWht_Miq6ULr3EVgq49hPEWH8bz4
```
Next Steps

- The people are waiting this...

  ... I’d ask that the WG consider a Call for Adoption