Pass the Cookie and Pivot to the Cloud

Disclaimer: Always make sure you have proper authorization before pen testing.

Johann Rehberger
wunderwuzzi23@outlook.com

December 2018
Cookies – Kekse - Biscotti

- Web Applications & Services use cookies to authenticate users.

- Single Key to the Kingdom
  - If you have the appropriate cookie => Access!
  - 2FA happened already, so that won’t protect you
    => probably often more valuable then your password.

- A cookie could be the single key to the “virtual” datacenter of your organization
- A cookie could be the single key to your personal finances, crypto, etc...
- ...

So, be aware of the value of cookies, protect and monitor them well!
What is Pass The Cookie?

Pass the Cookie is a post exploitation session hijacking technique.

After compromising a valuable host an adversary steals authentication cookies from browsers and related processes.

The adversary passes the acquired cookies to elevate privileges and pivot from the host to the corresponding cloud service.

This bypasses most multi-factor authentication protocols.

Successfully used during adversarial emulation to achieve mission objective.
Acquiring Cookies, Tools and Techniques

Here are some tools that can be leveraged (there are more):

- **firefox_creds** - Access SQL Lite Database of Firefox
- **Cookie Crimes** - Neat way to grab cookies via headless Chrome
- **ProcDump** - Swiss army knife to dump strings from any process

How to Pass the Cookie?

- Leverage default browser features
  E.g. using Developer Console (`document.cookie = “key=value”`) or UI

Disclaimer: Always make sure you have proper authorization before pen testing.
Example – Pass the Cookie

(1) Unauthenticated

(2) Pass the Cookie

(3) Refresh => Authenticated!

Disclaimer: Always make sure you have proper authorization before pen testing.
Anatomy of a Cloud Breach

A cookie might be the single key to your companies “virtual datacenter”.

- Attacker stole cookies of a GCP user.
- Navigates to root domain. Observe that we are unauthenticated.
- Pass the Cookies using the browser user interface
- Refresh the browser and observe being logged in, then browse to GCP.

Disclaimer: Always make sure you have proper authorization before pen testing.
# Pass the Cookie - Cheat Sheet

<table>
<thead>
<tr>
<th>Application</th>
<th>Cookie Name</th>
<th>Domain</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Web Services</td>
<td>aws-userInfo, aws-creds</td>
<td>.amazon.com</td>
<td><a href="https://console.aws.amazon.com">https://console.aws.amazon.com</a></td>
</tr>
<tr>
<td>Google Cloud Platform</td>
<td>OSID, HSID, SID, SSID, APISID, SAPISID, LSID</td>
<td>.google.com</td>
<td><a href="https://console.cloud.google.com">https://console.cloud.google.com</a>, Set OSID on console.cloud.google.com, Others on .google.com</td>
</tr>
<tr>
<td>Microsoft Online</td>
<td>ESTSAUTHPERSISTENT</td>
<td>.microsoftonline.com</td>
<td></td>
</tr>
<tr>
<td>Facebook for Work</td>
<td>c_user, xs</td>
<td>.facebook.com</td>
<td>Also works for regular Facebook</td>
</tr>
<tr>
<td>OneLogin</td>
<td>sub_session_onelogin.com</td>
<td>.onelogin.com</td>
<td></td>
</tr>
<tr>
<td>GitHub</td>
<td>user_session</td>
<td>.github.com</td>
<td></td>
</tr>
<tr>
<td>Hotmail, Calendar</td>
<td>RPSSecAuth</td>
<td>.live.com</td>
<td></td>
</tr>
<tr>
<td>Gmail</td>
<td>OSID, HSID, SID, SSID, APISID, SAPISID, LSID</td>
<td>.google.com</td>
<td><a href="https://mail.google.com">https://mail.google.com</a>, For basic mail first 4 are enough.</td>
</tr>
</tbody>
</table>

Disclaimer: Always make sure you have proper authorization before pen testing.

[https://wunderwuzzi23.github.io/blog/passthecookie.html#CheatSheet](https://wunderwuzzi23.github.io/blog/passthecookie.html#CheatSheet)
Detections

When it comes to detections a few things come to mind:

• Monitor for **applications that perform process dumps**
• Monitor for **access anomalies of cookie storage locations, databases,**
• Monitor for **unusual activity on critical web assets** (like cloud provider management consoles, etc,...)
• Monitor for **login anomalies** (location, time, unusual access patterns)
• **Leverage features that cloud providers offer!** (Threat Detection,...)
• **Perform adversarial emulations in your organization to test detection capabilities.**
• ...
Mitigations

• **Regularly delete cookies**, so they get removed from your machine
• Delete session cookies as well
• **Be the only Administrator on your own machine**
• **Browse sensitive/high value sites from isolated machines**
  ➢ Accessing your companies AWS account with admin privileges from a regular dev box is not a good idea
  ➢ Don’t use your work computer for personal things (do you trust your Administrators)
• **Separation of duties.**
• **Leverage features that cloud providers offer!** (IAM, RBAC, Firewall, Threat Detection,...)
• Require further authentication proof for sensitive operations
• Requiring client side certificates makes it more difficult to pass the cookie
Thanks!

Contact
wunderwuzzi23@outlook.com

Relevant Resources:
• https://wunderwuzzi23.github.io/blog/passthecookie.html
• https://www.owasp.org/index.php/Session_hijacking_attack