*The following security alert was issued by the Information Security Division of the Mississippi Department of ITS and is intended for State government entities. The information may or may not be applicable to the general public and accordingly, the State does not warrant its use for any specific purposes.*

**TLP: WHITE**
**Disclosure is not limited. Subject to standard copyright rules, TLP: WHITE information may be distributed without restriction.**
**http://www.us-cert.gov/tlp/**

**DATE(S) ISSUED:**
07/09/2019

**SUBJECT:**
Multiple Vulnerabilities in Mozilla Firefox Could Allow for Arbitrary Code Execution

**OVERVIEW:**
Multiple vulnerabilities have been discovered in Mozilla Firefox and Firefox Extended Support Release (ESR), the most severe of which could allow for arbitrary code execution. Mozilla Firefox is a web browser used to access the Internet. Mozilla Firefox ESR is a version of the web browser intended to be deployed in large organizations. Successful exploitation of the most severe of these vulnerabilities could allow for arbitrary code execution. Depending on the privileges associated with the user, an attacker could then install programs; view, change, or delete data; or create new accounts with full user rights. Users whose accounts are configured to have fewer user rights on the system could be less impacted than those who operate with administrative user rights.

**THREAT INTELLIGENCE:**
There are currently no reports of these vulnerabilities being exploited in the wild.

**SYSTEMS AFFECTED:**
- Mozilla Firefox versions prior to 68
- Mozilla Firefox ESR versions prior to 60.8

**RISK:**
**Government:**
- Large and medium government entities: **High**
- Small government entities: **Medium**

**Businesses:**
- Large and medium business entities: **High**
- Small business entities: **Medium**

**Home users: Low**

**TECHNICAL SUMMARY:**
Multiple vulnerabilities have been discovered in Mozilla Firefox and Firefox Extended Support Release (ESR), the most severe of which could allow for arbitrary code execution. Details of the vulnerabilities are as follows:

- Activity Stream can display content from sent from the Snippet Service website. This content is written to innerHTML on the Activity Stream page without sanitization, allowing for a potential access to other information available to the Activity Stream, such as browsing history, if the Snipper Service were compromised. (CVE-2019-11718)

- Application permissions give additional remote troubleshooting permission to the site input.mozilla.org, which has been retired and now redirects to another site. This additional permission is unnecessary and is a potential vector for malicious attacks. (CVE-2019-11724)
- A sandbox escape by installing a malicious language pack and then opening a browser feature that used the compromised translation. (CVE-2019-9811)
- A use-after-free vulnerability can occur in HTTP/2 when a cached HTTP/2 stream is closed while still in use, resulting in a potentially exploitable crash. (CVE-2019-11713)
- A vulnerability exists during the installation of add-ons where the initial fetch ignored the origin attributes of the browsing context. This could leak cookies in private browsing mode or across different "containers" for people who use the Firefox Multi-Account Containers Web Extension. (CVE-2019-11723)
- A vulnerability exists where if a user opens a locally saved HTML file, this file can use file: URIs to access other files in the same directory or sub-directories if the names are known or guessed. The Fetch API can then be used to read the contents of any files stored in these directories and they may uploaded to a server. It has determined that in combination with a popular Android messaging app, if a malicious HTML attachment is sent to a user and they opened that attachment in Firefox, due to that app's predictable pattern for locally-saved file names, it is possible to read attachments the victim received from other correspondents. (CVE-2019-11730)
- A vulnerability exists where it possible to force Network Security Services (NSS) to sign CertificateVerify with PKCS#1 v1.5 signatures when those are the only ones advertised by server in CertificateRequest in TLS 1.3. PKCS#1 v1.5 signatures should not be used for TLS 1.3 messages. (CVE-2019-11727)
- A vulnerability exists where the caret ("^") character is improperly escaped constructing some URIs due to it being used as a separator, allowing for possible spoofing of origin attributes. (CVE-2019-11717)
- Due to an error while parsing page content, it is possible for properly sanitized user input to be misinterpreted and lead to XSS hazards on web sites in certain circumstances. (CVE-2019-11715)
- Some memory safety bugs present in Firefox 67 showed evidence of memory corruption and we presume that with enough effort that some of these could be exploited to run arbitrary code. (CVE-2019-11710)
- Some memory safety bugs present in Firefox 67 and Firefox ESR 60.7 showed evidence of memory corruption and we presume that with enough effort that some of these could be exploited to run arbitrary code. (CVE-2019-11709)
- Empty or malformed p256-ECDH public keys may trigger a segmentation fault due values being improperly sanitized before being copied into memory and used. (CVE-2019-11729)
- Necko can access a child on the wrong thread during UDP connections, resulting in a potentially exploitable crash in some instances. (CVE-2019-11714)
- POST requests made by NPAPI plugins, such as Flash, that receive a status 308 redirect response can bypass CORS requirements. This can allow an attacker to perform Cross-Site Request Forgery (CSRF) attacks. (CVE-2019-11712)
- Some unicode characters are incorrectly treated as whitespace during the parsing of web content instead of triggering parsing errors. This allows malicious code to then be processed, evading cross-site scripting (XSS) filtering. (CVE-2019-11720)
- The HTTP Alternative Services header, Alt-Svc, can be used by a malicious site to scan all TCP ports of any host that the accessible to a user when web content is loaded. (CVE-2019-11728)

- The unicode latin 'kra' character can be used to spoof a standard 'k' character in the addressbar. This allows for domain spoofing attacks as do not display as punycode text, allowing for user confusion. (CVE-2019-11721)
- Until explicitly accessed by script, window.globalThis is not enumerable and, as a result, is not visible to code such as Object.getOwnPropertyNames(window). Sites that deploy a sandboxing that depends on enumerating and freezing access to the window object may miss this, allowing their sandboxes to be bypassed. (CVE-2019-11716)
- When an inner window is reused, it does not consider the use of document.domain for cross-origin protections. If pages on different subdomains ever cooperatively use document.domain, then either page can abuse this to inject script into arbitrary pages on the other subdomain, even those that did not use document.domain to relax their origin security. (CVE-2019-11711)
- When a user navigates to site marked as unsafe by the Safebrowsing API, warning messages are displayed and navigation is interrupted but resources from the same site loaded through websockets are not blocked, leading to the loading of unsafe resources and bypassing safebrowsing protections. (CVE-2019-11725)
- When importing a curve25519 private key in PKCS#8format with leading 0x00 bytes, it is possible to trigger an out-of-bounds read in the Network Security Services (NSS) library. This could lead to information disclosure. (CVE-2019-11719)

Successful exploitation of the most severe of these vulnerabilities could allow for arbitrary code execution. Depending on the privileges associated with the usergrep  an attacker could then install programs; view, change, or delete data; or create new accounts with full user rights. Users whose accounts are configured to have fewer user rights on the system could be less impacted than those who operate with administrative user rights.

**RECOMMENDATIONS:**
The following actions should be taken:
- Apply appropriate updates provided by Mozilla to vulnerable systems, immediately after appropriate testing.
- Run all software as a non-privileged user (one without administrative privileges) to diminish the effects of a successful attack.
- Remind users not to visit un-trusted websites or follow links provided by unknown or un-trusted sources.
- Inform and educate users regarding the threats posed by hypertext links contained in emails or attachments especially from un-trusted sources.
- Apply the Principle of Least Privilege to all systems and services.

**REFERENCES:**
**Mozilla:**
https://www.mozilla.org/en-US/security/advisories/mfsa2019-21/
https://www.mozilla.org/en-US/security/advisories/mfsa2019-22/

**CVE:**
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-9811
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11709
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11710
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11711
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11712
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11713

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11714
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11715
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11716
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11717
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11718
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11719
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11720
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11721
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11723
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11724
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11725
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11727
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11728
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11729
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11730



**Chris Watts**
Security Operations Analyst
MS Department of Information Technology Services
601-432-8201 | www.its.ms.gov