1.1.1 Önbelleklenebilir HTTPS Yanıtı (Cacheable HTTPS Response) (CWE-16)

Açıklık Önem Derecesi: Düşük

Açıklığın Etkisi: Bilgi İfşası

Açıklığın Barındıran Dosyalar/Satırlar:

Proje Dosyası/Dosya Adı	Satır Numarası

Açıklığın Açıklaması:

Son kullanıcılar sistemlerindeki web tarayıcı yazılımları aracılığıyla çeşitli web sitelerine (sunucularına) talepte bulunduklarında web sunuculardan dönen https yanıtlarını bir klasör altında toplayabilmektedirler. Örneğin eğer ziyaret edilen web sunucular gönderdikleri https yanıt paketlerini önbelleklenebilirlik açısından bir yapılandırmaya tabi tutmamışlarsa hizmet olarak sunulan web sitelerinin bazı parçalarını (örn; html, css, js, jpg, php'nin html çıktısı, aspx'in html çıktısı... gibi) son kullanıcı bilgisayarının web tarayıcı önbelleğine bırakabilirler. Bu prosedür son kullanıcının daha hızlı / daha performanslı bir şekilde web sitesi içerisinde gezinebilmesini ve değişmeyen parçalar için her defasında sürekli aynı kaynağın web sunucudan indirilmesi gereksizliğinin önlenmesiyle web sunucudaki yükün (web sunucu trafiğinin) rahatlatılmasını sağlar. Fakat bu avantajlar aynı zamanda bazı riskler de doğurur.

Son kullanıcı, bilgisayarındaki web tarayıcıda HTTPS protokolü kullanan bir web uygulamasını kullandığında paketlerini HTTPS protokolü aracılığıyla uçtan uca şifreli bir şekilde güvenle karşıya aktarır ve karşıdan güvenle paketlerini alır. Böylece kritik mahiyette göndereceği verilerin aradaki saldırganlarca okunabilmesinin önüne geçer. Fakat web sunucu kendisine gönderilen https taleplerine karşılık döndüğü https yanıtlarını eğer önbelleklenebilir şekilde gönderiyorsa https yanıt paketleri - aktarım sırasında her ne kadar meraklı gözlerce korunuyor olsa da - son kullanıcıya ulaştığında son kullanıcı bilgisayarında şifresi çözülerek ekrana yansıtılacağından şifrelenmemiş hali kullanıcı sistemindeki tarayıcı önbelleğine kaydedilecektir. Bu ise iki türlü risk teşkil edecektir. Birincisi ilgili bilgisayara gelecekte fiziksel erişimi olan bir saldırgan web tarayıcıyı açarak önbellekten gelen https sayfaları görüntüleyerek yetkisizce bilgileri ele geçirebilir. İkincisi bilgisayarda yazılımsal olarak olası ciddi bir zafiyetin keşfi sonrası sisteme RDP (Uzak Masaüstü Bağlantısı) ile sızacak saldırgan yine web tarayıcıyı açarak önbellekten gelen https sayfaları görüntüleyerek yetkisizce bilgileri ele geçirebilir. Dolayısıyla HTTPS yanıtlarının son kullanıcı makinalarındaki tarayıcılarda önbelleğe kaydedilmesinin önüne geçilmesi önerilmektedir.

Kurum web sunucusunun kullanıcılara döndüğü https yanıtlarının önbelleklenebilir şekilde istemcilere (kullanıcılara) gönderdiği tespit edilmiştir:

[[[[[[[[[[[[[[BULGU]]]]]]]]]]]]]]

Açıklığın Önlemi:

Web sunucusu HTTPS protokolüyle dahi paketleri gönderiyor olsa da paketler sonuçta istemcinin görüntülemesi sırasında açık metin haline geldiğinden şayet paketler önbelleklenebilir durumda bırakılmışsa paketlerin istemci makinadaki tarayıcı önbelleğine kaydolacağını göz önünde bulundurarak bu riski bertaraf etmek için web sunucu, kendi içerisinde yapısal bir değişikliğe gitmelidir ve göndereceği her https yanıt paketine

```
Cache-control: no-store
Pragma: no-cache
```

başlıklarını ilave etmelidir. Bu sayede kullanıcının girebileceği kritik verilerin kullanıcı sistemindeki tarayıcı önbelleğine kaydedilmesinin önüne geçilmiş olacaktır.

Pragma HTTP/1.0 çözümüdür ve artık tedavülden kalkmıştır. HTTP/1.1 için yeni ve daha kapsamlı seçenekler sunan Cache-control çözümü kullanılmaktadır. Fakat HTTP/1.1'i desteklemeyen eski istemciler (tarayıcılar) için Pragma halen ilave olarak kullanılabilir. Böylece web uygulamanın geriye dönük, teknolojisi eski kimselere olan desteği sürecektir.

İstemci tarafında HTTPS paketlerinin tarayıcı önbelleğine kaydedilmesini engelleme ve böylece kullanıcıların bilgi güvenliğini sağlama yolu şu şekildedir:

a) IIS Web Sunucular

IIS sunucularda konfigürasyon dosyası Web.config açılmalıdır ve httpprotocol etiketi içerisindeki customheaders etiketi içerisine gösterilen satırlar eklenmelidir.

```
<!-- GÜVENLİ YAPILANDIRMA -->

<?xml version="1.0" encoding="utf-8"?>
<configuration>
    <system.webServer>
        <security>
            ...
        </security>
        <httpProtocol>
            <customHeaders>
                <add name="Cache-Control" value="no-store ">
                </add>
                <add name="Pragma" value="no-cache ">
                </add>
            </customHeaders>
        </httpProtocol>
    </system.webServer>
</configuration>
```

Ardından IIS servisi konsoldan veya arayüzden yeniden başlatılmalıdır.

CMD (Yönetici Olarak) Konsolu:

```
> iisreset
```

IIS Yöneticisi Arayüz Paneli:

```
- IIS Manager panelinde yeniden başlat seçeneğine tıklanır.
```

b) Apache Web Sunucular

Apache sunucular için Debian tabanlı sistemlerde apache2.conf dosyası, RedHat / Centos tabanlı sistemlerde httpd.conf dosyası açılmalıdır ve dosya içeriğinin en altına belirtilen satırlar eklenmelidir.

Terminal:

```
> sudo su
> a2enmod headers
> nano /etc/apache2/apache2.conf
```

Çıktı:

```
...

(( eklenecek satırlar ))

<filesMatch ".*$">
Header set Cache-Control "no-store"
Header set Pragma "no-cache"
```

Ardından apache servisi uygun komutla yeniden başlatılır:

```
// Debian tabanlı sistemlerde
> service apache2 restart

// RedHat / Centos tabanlı sistemlerde
> service httpd restart
```

Apache konfigürasyon dosyasına eklenen satırlar ile kurum sunucusunun tüm https yanıtlarında gönderdiği bileşenlerin istemci makinasında depolanmasının önüne geçilmiş olur. Fakat tercihe göre bu katı kural kullanıcıların web uygulamasını görüntüleme hızını / performansını arttırmak için biraz esnetilebilir. Örneğin konfigürasyon dosyası eklenen satırlar şu şekilde düzenlenecek olursa

Terminal:

```
> nano /etc/apache2/apache2.conf
```

Çıktı:

```
(( eklenecek satırlar ))

<filesMatch ".(html|htm|js|css)$">
Header set Cache-Control "no-store"
Header set Pragma "no-cache"
```

sadece html, htm, js ve css uzantılı dosya gönderen https yanıtları istemci makinelerinde depolanmayacaktır. Geri kalan uzantılar (örn; png, jpeg, flv, swf,..) https yanıtlarıyla giderken

Cache-Control ve Pragma başlıklarıyla gitmeyeceği için istemci makinesine depolanabilecektir. Bunun gibi ayrıca tek tek hangi uzantılardaki dosyaları içeren https yanıtlarının istemci makinelerinde depolanmayacağı da belirtilebilir. Örn;

Terminal:

```
> nano /etc/apache2/apache2.conf
```

Çıktı:

```
...

(( eklenecek satırlar ))

<filesMatch ".(html|htm|js|css|ico|pdf|flv|jpg|jpeg|png|gif|swf)$">
Header set Cache-Control "no-store"
Header set pragma "no-cache"
```

Referanslar:

1. https://cwe.mitre.org/data/definitions/16
2. https://portswigger.net/kb/issues/00700100_cacheable-https-response
3. https://stackoverflow.com/questions/11532636/how-to-prevent-http-file-caching-in-apache-httpd-mamp
4. https://stackoverflow.com/questions/10314174/difference-between-pragma-and-cache-control-headers
5. https://www.keycdn.com/blog/http-cache-headers
6. https://stackoverflow.com/questions/36783181/how-can-i-apply-clientcache-settings-to-a-specific-extension-in-iis
7. https://forums.iis.net/t/1195180.aspx