年度最慘漏洞！深入探究 Oracle WebLogic CVE-2020-14882

teamt5.org/tw/posts/most-epic-fail-vulnerability-research-on-oracle-weblogic-cve-2020-14882

Sonar Team

1.13.2021Sonar Team

Share:

緣起

WebLogic 是美商 Oracle 的主要產品之一，係商業市場上主要的 Java（J2EE）應用伺服器軟體（application server）之一。於 2020 年爆出高風險漏洞 CVE-2020-14882 與 CVE-2020-14883，其中，CVE-2020-14882 的 CVSS 3.1 評分更高達 9.8，屬於嚴重（Critical）等級之漏洞。

攻擊者可利用漏洞，組合惡意指令，使未經授權的請求繞過 WebLogic 後台登入等限制，最終可以遠端執行代碼攻擊（Remote Code Execution），因此攻擊者可輕易的利用此漏洞發起攻擊。

技術分析

我們從下圖的漏洞 POC 範例開始看起。

階段 A　　階段 B　　階段 C

http://127.0.0.1:7001/console/images/%252E%252E%252Fconsole.portal?_nfpb=false&_pageLabel=&handle=com.tangosol.coherence.mvel2.sh.ShellSession(%22java.lang.Runtime.getRuntime().exec(%27whoami%27);%22);

階段 D　　階段 E　　階段 F

圖一、POC 攻擊範例

首先 WebLogic 會對 url 進行檢查，當攻擊者試圖存取 console 路徑底下的檔案資源時（階段 A），會將其 url 與 WebLogic 設定檔（web.xml）內的網頁資源路徑進行比對，若該 url 屬於任一資源路徑，將不須驗證即可進行資源存取（階段 B）。其 web.xml 的檔案路徑為 wlserver/server/lib/consoleapp/webapp/WEB-INF/web.xml，如下圖所示。

```
<!-- Static resources that don't need protection -->
<security-constraint>
  <web-resource-collection>
    <web-resource-name>images</web-resource-name>
    <url-pattern>/images/*</url-pattern>
  </web-resource-collection>
  <web-resource-collection>
    <web-resource-name>common</web-resource-name>
    <url-pattern>/common/*</url-pattern>
  </web-resource-collection>
  <web-resource-collection>
    <web-resource-name>css</web-resource-name>
    <url-pattern>/css/*</url-pattern>
    <url-pattern>/framework/skeletons/wlsconsole/css/*</url-pattern>
    <url-pattern>/framework/skeletons/wlsconsole/js/*</url-pattern>
    <url-pattern>/framework/skins/wlsconsole/css/*</url-pattern>
    <url-pattern>/framework/skins/wlsconsole/images/*</url-pattern>
  </web-resource-collection>
</security-constraint>

 <security-constraint>
  <web-resource-collection>
    <web-resource-name>helpsets</web-resource-name>
    <url-pattern>/bea-helpsets/*</url-pattern>
  </web-resource-collection>
</security-constraint>
```

圖二、web.xml 內容

再來，`http://127.0.0.1:7001/console/images/%252E%252E%252Fconsole.portal` 會先進行一次 url decode，將 %25 轉成 %，所以該 url 會變成 `http://127.0.0.1:7001/console/images/%2E%2E%2Fconsole.portal`（階段 C）。

但是在 UIServletInternal.getTree 函式中，WebLogic 又會做再一次 url decode，將 %2E%2E%2F 轉為 ../，於是 url 變成 `http://127.0.0.1:7001/console/images/../console.portal`，且後續的操作中 WebLogic 都沒有對路徑進行權限檢查，所以產生了跨目錄穿越攻擊（Directory Traversal）的漏洞。

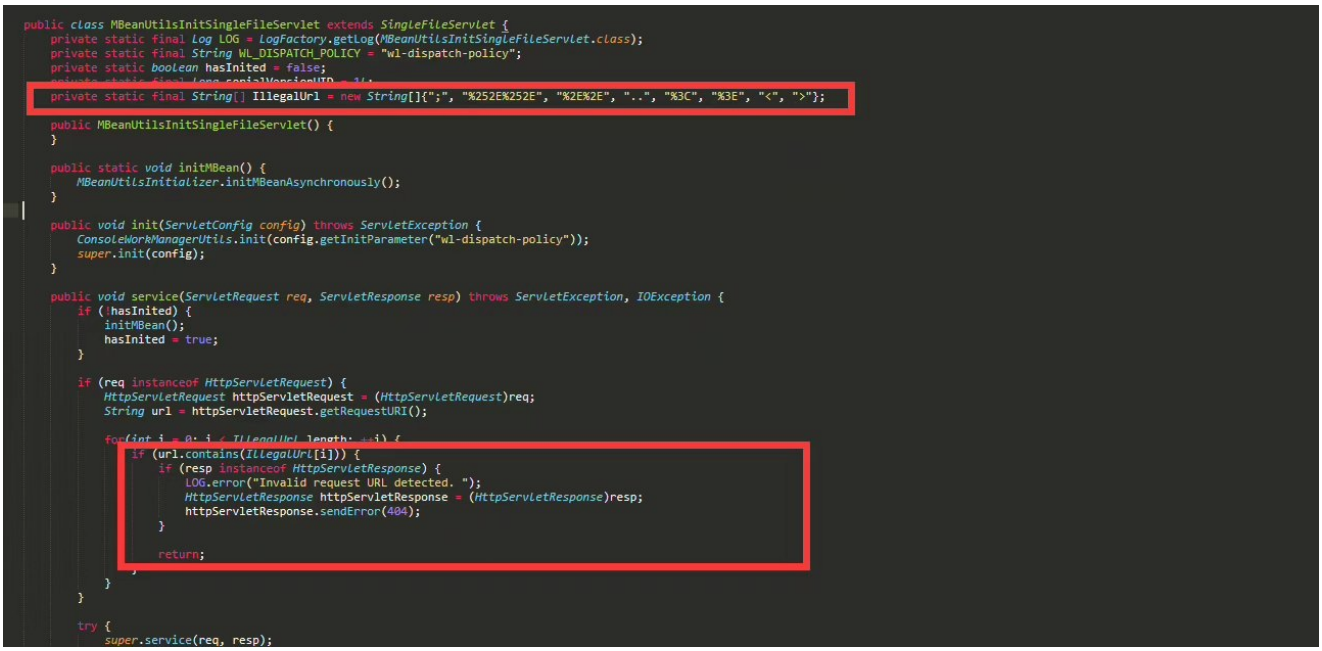
WebLogic Server 處理後台管理（console）的請求時，會呼叫 BreadcrumbBacking().init 函式進行處理（階段 D）。當 url 中含有 handle 參數時，會將 handle 拆解成 class 和 argument。以 POC 為例，class 就會是 `com.tangosol.coherence.mvel2.sh.ShellSession()`（階段 E），argument 則是 `"java.lang.Runtime.getRuntime().exec(%27whoami%27);"`（階段 F）。

後續，會將 argument 傳給 class 的 constructor ，而 ShellSession 函式會再把 argument 傳給 MVELInterpretedRuntime 函式中，最後透過反射執行 argument 中內含的惡意指令，也就是 POC 範例中的 whoami。

修補插曲

但僅憑跨目錄穿越攻擊、Unauthorized RCE 等問題，何以稱之為「年度最慘漏洞（Most Epic Fail）」？

Oracle 於 2020 年 10 月釋出修補程式，但細究其修補的方法，發現 Oracle 利用黑名單的方式過濾 url 參數，如下圖所示。

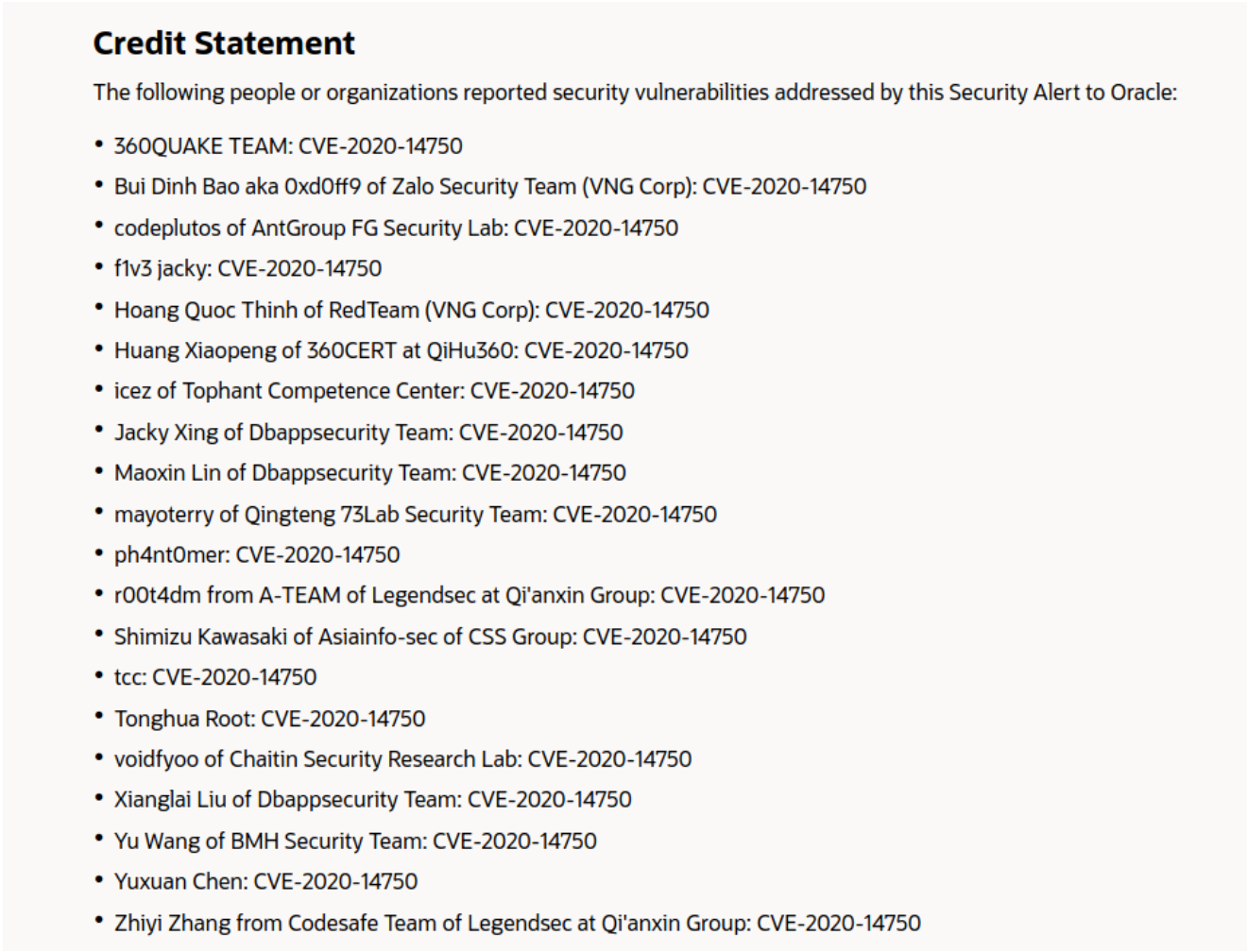


圖三、Oracle 透過黑名單修補 CVE-2020-14882

利用黑名單方式過濾，會有什麼問題呢？

只要利用不在黑名單內的 %252e%252e，就可以成功繞過（Bypass）已修補程式中的 url 檢查。

眾多安全研究人員於短時間內發現該問題並通報 Oracle，該漏洞也被編號為 CVE-2020-14750，如下圖所示。此錯誤的漏洞修補方式，也使得 CVE-2020-14882 成功入圍 2020 年 Pwnie Award Most Epic Fail 項目。



Credit Statement

The following people or organizations reported security vulnerabilities addressed by this Security Alert to Oracle:

- 360QUAKE TEAM: CVE-2020-14750
- Bui Dinh Bao aka 0xd0ff9 of Zalo Security Team (VNG Corp): CVE-2020-14750
- codeplutos of AntGroup FG Security Lab: CVE-2020-14750
- f1v3 jacky: CVE-2020-14750
- Hoang Quoc Thinh of RedTeam (VNG Corp): CVE-2020-14750
- Huang Xiaopeng of 360CERT at QiHu360: CVE-2020-14750
- icez of Tophant Competence Center: CVE-2020-14750
- Jacky Xing of Dbappsecurity Team: CVE-2020-14750
- Maoxin Lin of Dbappsecurity Team: CVE-2020-14750
- mayoterry of Qingteng 73Lab Security Team: CVE-2020-14750
- ph4nt0mer: CVE-2020-14750
- r00t4dm from A-TEAM of Legendsec at Qi'anxin Group: CVE-2020-14750
- Shimizu Kawasaki of Asiainfo-sec of CSS Group: CVE-2020-14750
- tcc: CVE-2020-14750
- Tonghua Root: CVE-2020-14750
- voidfyoo of Chaitin Security Research Lab: CVE-2020-14750
- Xianglai Liu of Dbappsecurity Team: CVE-2020-14750
- Yu Wang of BMH Security Team: CVE-2020-14750
- Yuxuan Chen: CVE-2020-14750
- Zhiyi Zhang from Codesafe Team of Legendsec at Qi'anxin Group: CVE-2020-14750

圖四、眾多研究員回報 CVE-2020-14750

漏洞修補是一門深奧的學問，以 CVE-2020-14882 為例，其成因包含對檔案路徑存取控管不確實、對路徑做兩次的 url decode 及後台管理程式提供代碼執行。但 Oracle 一開始只修補兩次的 url decode 問題，更不用說其修補的方式為黑名單限制。因漏洞修補不確實而產生了更多的漏洞，所以不可不慎。

影響層面

攻擊者利用此漏洞互相搭配，不須登入即可遠端指令執行，進而產生憑證竊取（Credential Dump）、橫向移動（Lateral Movement）等攻擊行為。

- 受影響產品：WebLogic Server
- 受影響版本：10.3.6.0.0、12.1.3.0.0、12.2.1.3.0、12.2.1.4.0、14.1.1.0.0

防護建議

- 更新 Oracle 官方最新修正 patch。
- 惡意 payload 檢查，例如檢查 console.portal 和 payload 是否帶有 `handle=...` 參數，或是直接關閉 console.portal 和 payload 的 handle 功能。

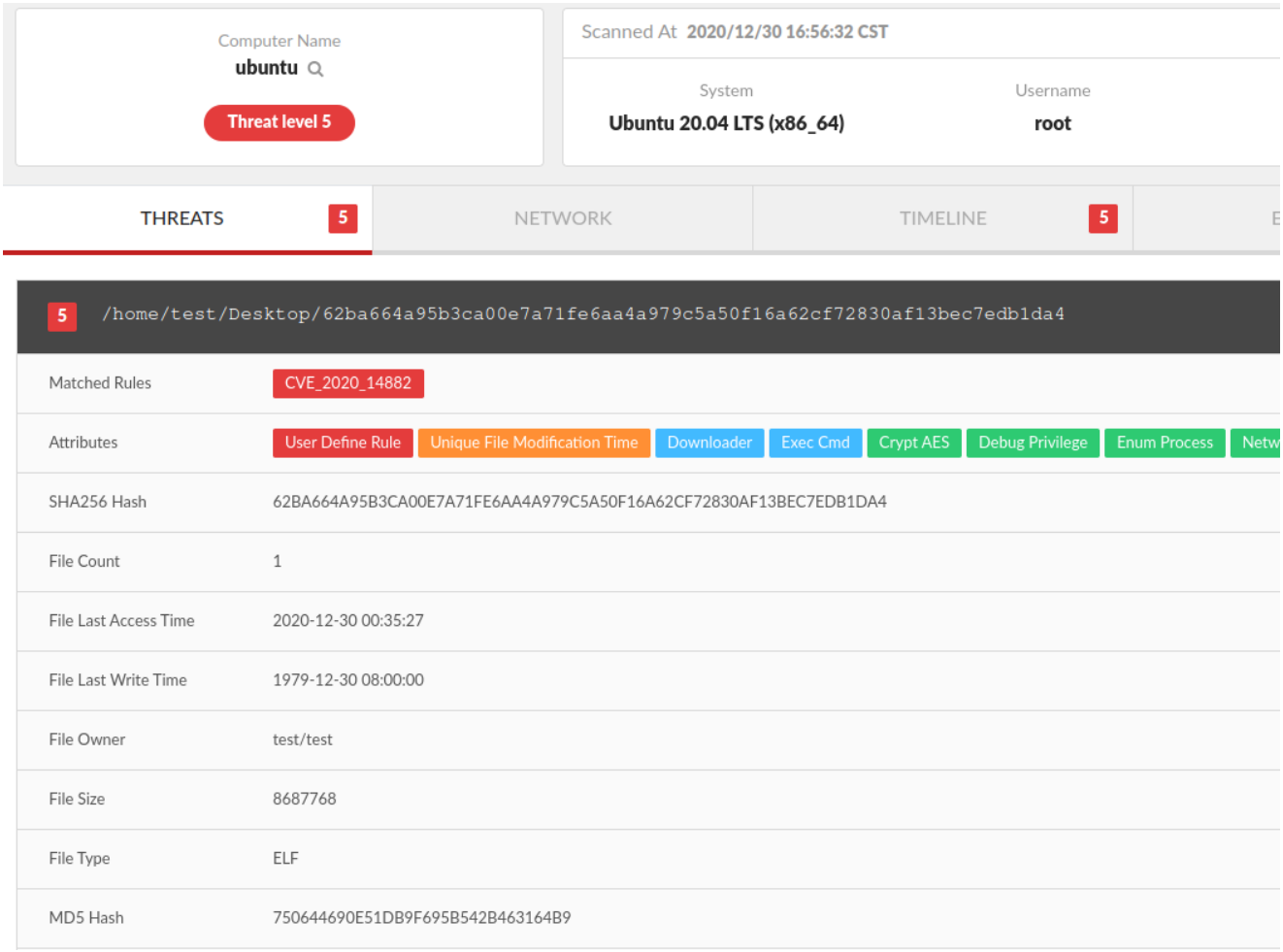
- 匯入以下 Yara 偵測規則，確認是否存在利用該漏洞的惡意程式。

```
rule CVE_2020_14882 {
    meta:
        author = "TeamT5"
        description = "CVE-2020-14882 exploit"
    strings:
        $resource1 = "console/bea-helpsets/"
        $resource2 = "console/framework/skins/wlsconsole/images/"
        $resource3 = "console/framework/skins/wlsconsole/css/"
        $resource4 = "console/framework/skins/wlsconsole/js/"
        $resource5 = "console/framework/skeletons/wlsconsole/css/"
        $resource6 = "console/framework/skeletons/wlsconsole/js/"
        $resource7 = "console/css/"
        $resource8 = "console/common/"
        $resource9 = "console/images/"

        $console1 = "console.portal"
        $console2 = "consolejndi.portal"

        $handle = "handle="

    condition:
        any of ($resource*) and 1 of ($console*) and $handle
}
```



圖五、透過 Yara 偵測規則可有效偵測 CVE-2020-14882 exploit

*圖片來源：Pixabay

Share: