Web & Crypto

任务驱动的ICS技术学习

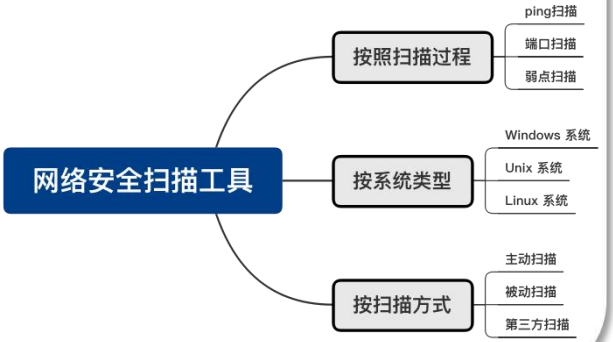
2024 年 6 月 5 日

Web——扫描工具

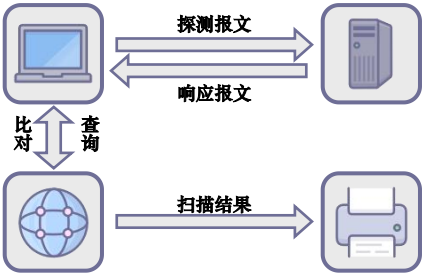
任务驱动的ICS技术学习

网络扫描

扫描工具基本分类

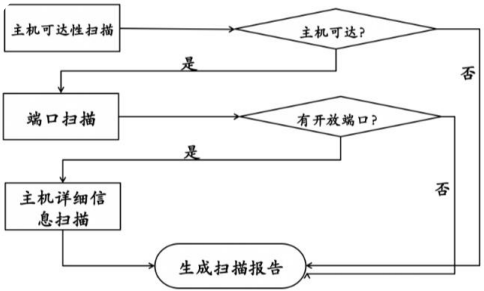


基本思想



首先进行**查询**即扫描知识库，**构建探测报文**；
再向目标主机**发送探测报文**，
然后**接受目标响应报文**并扫描知识库，**比对响应报文**；
最后生成扫描结果报文

Nmap——简介



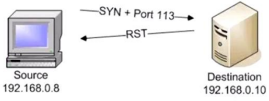
Nmap扫描流程

- Nmap是**主动扫描工具**，它会向目标主机发送探针，根据目标主机的回应猜测系统。这种探针大都是**TCP**和**UDP**数据包的形式。
- Nmap功能非常强大，具有**主机存活检测、端口扫描、网络服务及版本辨识、操作系统检测、漏洞扫描**分析五项核心功能。
- Nmap可以运行在所有主流的操作系统上，如**Linux**、**Windows**和**Mac OS**等。
- 相比较一般扫描器对端口分为开放或关闭两种类型，Nmap对端口的分析粒度更加细致，共分为**开放、关闭、被过滤**等多类状态。

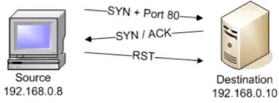
浙江大学 ZHEJIANG UNIVERSITY

Nmap——基本扫描方式

TCP SYN scanning

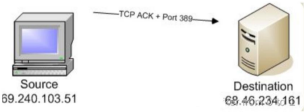


TCP SYN 探测到端口关闭

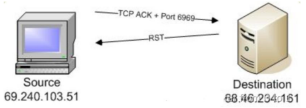


TCP SYN 探测到端口打开

TCP ACK scanning

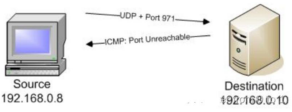


TCP ACK 探测到端口被屏蔽

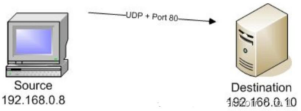


TCP ACK 探测到端口未被屏蔽

UDP scanning



UDP 端口关闭



UDP 端口开放或被屏蔽

浙江大学 ZHEJIANG UNIVERSITY

Nmap——基本指令

命令行参数	说明
-sT	TCP connect() 扫描，这种方式会在目标主机的日志中记录大批连接请求和错误信息。
-sS	半开扫描，很少有系统能把它记入系统日志。不过，需要 Root 权限。
-sS -sN	秘密 FIN 数据包扫描、Xmas Tree、Null 扫描模式
-sP	ping 扫描，Nmap 在扫描端口时，默认都会使用 ping 扫描，只有主机存活，Nmap 才会继续扫描。
-sU	UDP 扫描，但 UDP 扫描是不可靠的
-sA	这项高级的扫描方法通常用来穿过防火墙的规则集
-sV	探测端口服务版本
...	...

允许用户编写脚本进行自动化扫描操作,或者扩展Nmap现有的功能脚本

Nmap——基本指令

```
如：nmap --script=brute 192.168.0.101
```

nmap --script=brute ip 可以对数据库、MB、SNMP等进行简单的暴力破解

```
如：nmap --script=vuln 192.168.0.101
```

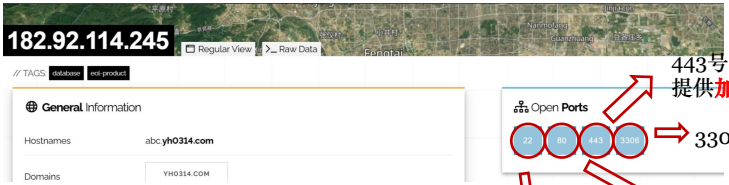
nmap --script=vuln ip 扫描常见漏洞

```
如：nmap -script external baidu.com
```

nmap -script external url 进行whois解析

允许用户编写脚本进行自动化扫描操作,或者扩展Nmap现有的功能脚本

Nmap——扫描实战



https://www.shodan.io/host/182.92.114.245

443号端口：
提供**加密**和通过**安全端口**传输的另一种 **HTTP**服务

3306号端口：**MySQL**数据库服务的标准端口

80号端口：**HTTP服务**的标准端口

22号端口：**SSH服务**的默认端口

执行服务版本探测和操作系统类型识别

各开放端口状态，服务与支持协议等

```
┌──(yzl㉿kali)-[~/Desktop]
└─$ sudo nmap -sV -O 182.92.114.245
[sudo] password for yzl:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-29 14:31 CST
Nmap scan report for 182.92.114.245
Host is up (0.025s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 7.4 (protocol 2.0)
80/tcp   open  http     nginx
443/tcp  open  ssl/http nginx
3306/tcp open  mysql    MySQL 5.6.50-log
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Aggressive OS guesses: DD-WRT v24-sp2 (Linux 2.4.37) (97%), Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012 (
97%), Actiontec MI424WR-GEN3I WAP (96%), VMware Player virtual NAT device (96%), Microsoft Windows XP SP3 (95%), Linux 3.2
(93%), Linux 4.4 (93%), BlueArc Titan 2100 NAS device (90%)
No exact OS matches for host (test conditions non-ideal).
```

按概率列出几个可能性比较高的操作系统匹配

Nmap——扫描实战

使用特定脚本ssh-hostkey来获取该SSH服务的主机密钥信息

Nmap——扫描实战

扫描潜在漏洞

```
┌──(yzl㉿kali)-[~/Desktop]
└─$ nmap --script=vuln 182.92.114.245
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-29 13:38 CST
443/tcp  open  https
|_http-csrf: Couldn't find any CSRF vulnerabilities.
| http-vuln-cve2011-3192:
|   VULNERABLE:
|   Apache byterange filter DoS
|     State: VULNERABLE
|     IDs:  BID:49303  CVE:CVE-2011-3192
|       The Apache web server is vulnerable to a denial of service attack
|       overlapping byte ranges are requested.
|     Disclosure date: 2011-08-19
|     References:
|       https://seclists.org/fulldisclosure/2011/Aug/175
|       https://www.tenable.com/plugins/nessus/55976
|       https://www.securityfocus.com/bid/49303
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2011-3192
```

CVE-2011-3192:

“Apache byterange filter Dos”

字节范围过滤器拒绝服务

这个漏洞意味着攻击者可以通过发送精心构造的HTTP请求，利用Apache服务器的字节范围过滤器，可能导致服务器资源耗尽，从而导致服务中断。

P0f——简介

功能

- 高度可扩展性
- 能够快速识别一个TCP连接两端的主机操作系统
- 探测系统运行时间，网络连接,距离
- 自动探测NAT，负载均衡，应用级的代理设置情况

原理

•对于TCP/IP，该工具对客户机发起的SYN包和来自服务器的第一个SYN+ACK响应进行指纹识别，并注意诸如TCP选项的顺序、最大段大小和窗口大小之间的关系、TCP时间戳的进展等因素。

•用于应用程序级流量的指标因模块而异。在可能的情况下，该工具依赖于如HTTP头部或SMTP命令行的排序或语法之类的信号，而不是声明性语句，如用户代理。

P0f用于被动获取操作系统指纹以识别远程主机操作系统

P0f——抓包分析

命令行参数语法：p0f [选项] [过滤规则]

-i <iface>	指定监听的网络接口
-r <file>	读取由抓包工具抓到的网络数据包文件
-p-	设置 -i参数 指定的网卡 为混杂模式
-L-	列出所有可用接口
-f <file>	指定指纹数据库 (p0f.fp) 路径，不指定则使用默认数据库
-o <file>	将信息写入指定的日志文件中
-s <name>	回答 unix socket 的查询 API
-u <user>	以指定用户身份运行程序，工作目录会切换到到当前用户根目录下；
-d	以后台进程方式运行p0f ,需要配合-0或者-s选项

P0f捕获IPv4和IPv6头、TCP头、TCP握手以及应用层的数据

P0f——抓包分析

嗅探准备

p0f

表示已经开始嗅探
如果打开浏览器就可以看到终端捕获的信息

```
┌──(root㉿kali)-[~]
└─# p0f
── p0f 3.09b by Michal Zalewski <lcamtuf@coredump.cx> ──

[+] Closed 1 file descriptor.
[+] Loaded 322 signatures from '/etc/p0f/p0f.fp'.
[+] Intercepting traffic on default interface 'eth0'.
[+] Default packet filtering configured [+VLAN].
[+] Entered main event loop.
```

p0f -L

列出所有可用接口
在这个例子里可以使用的接口只有eth0

```
┌──(root㉿kali)-[~]
└─# p0f -L
── p0f 3.09b by Michal Zalewski <lcamtuf@coredump.cx> ──


-- Available interfaces --

 0: Name        : eth0
    Description : -
    IP address  : 192.168.215.128

 1: Name        : any
    Description : Pseudo-device that captures on all interfaces
    IP address  : (none)

 2: Name        : lo
    Description : -
    IP address  : 127.0.0.1

 3: Name        : bluetooth0
    Description : Bluetooth adapter number 0
    IP address  : (none)
```

P0f——嗅探

p0f -i eth0 嗅探流经eth0接口的流量来识别连接双方的指纹信息

```
┌──(root㉿kali)-[~]
└─# p0f -i eth0
── p0f 3.09b by Michal Zalewski <lcamtuf@coredump.cx> ──

[+] Closed 1 file descriptor.
[+] Loaded 322 signatures from '/etc/p0f/p0f.fp'.
[+] Intercepting traffic on interface 'eth0'.
[+] Default packet filtering configured [+VLAN].
[+] Entered main event loop.

.-[ 192.168.215.128/42122 → 120.253.253.33/443 (syn) ]-
|
| client   = 192.168.215.128/42122
| os       = Linux 2.2.x-3.x
| dist     = 0
| params   = generic
| raw_sig  = 4:64+0:0:1460:mss*22,7:mss,sok,ts,nop,ws:df,id+:0
|
`----

.-[ 192.168.215.128/42122 → 120.253.253.33/443 (mtu) ]-
|
| client   = 192.168.215.128/42122
| link     = Ethernet or modem
| raw_mtu  = 1500
|
`----
```

从输出中获得信息

客户主机

- ip为192.168.215.128
- 端口号为42122
- 操作系统为Linux 2.2.x-3.x

访问主机

- ip为120.253.253.33
- 端口号为443
- 操作系统未识别

最大传输单元（MTU）为1500字节

POf——抓包分析

更多指纹信息

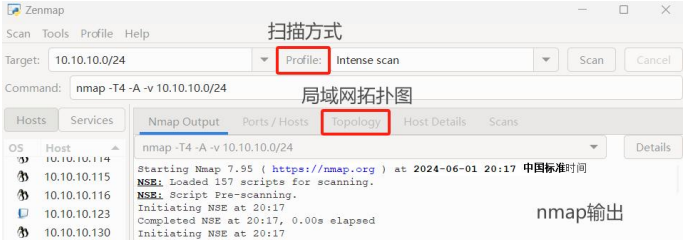
raw_sig:sig = ver : ittl : olen : mss : wsize , scale : olayout : quirks : pclass

第一栏	**协议**	表示使用的是IPv4还是IPv6协议
第二栏	**TTL**	表示IP首部的TTL值（每经过一个路由，TTL值减一）
第三栏	**扩展头长度**	表示IPV4的选项字段或者IPV6的扩展头部的长度
第四栏	**MTU**	表示最大报文段长度为1460个字节
第五栏	**窗口大小**	表示TCP的窗口大小为mss*22，窗口比例因子为7
第六栏	**olayout**	逗号分隔布局和排列TCP选项部分
第七栏	**quirks**	逗号分隔IP或TCP头部的特性和观察到的特殊情况
第八栏	**pclass**	有效载荷大小分类

Zenmap: nmap的用户图形界面

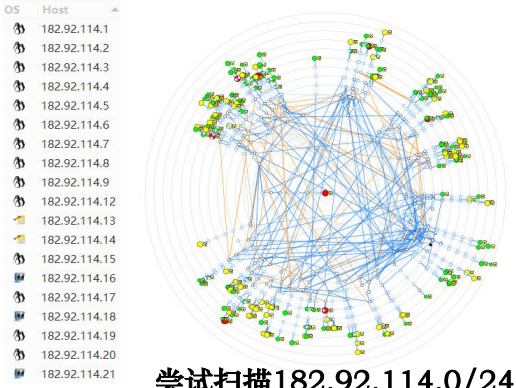
Zenmap提供了

直观的用户图形界面，
支持扫描结果的可视化，
扫描结果的保存和脚本引擎



扫描工具拓展

Zenmap 绘制局域网拓扑图



尝试扫描182.92.114.0/24

可视化网络连接方式

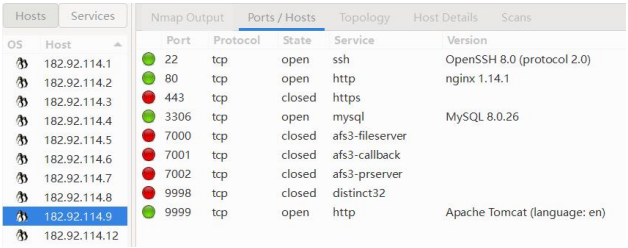
辅助规划网络结构

快速识别安全风险

快速定位风险来源

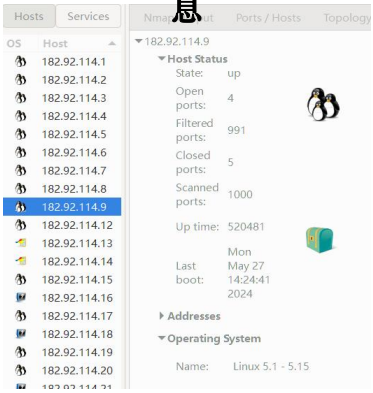
扫描工具拓展

更便捷 查看开放端口、服务情况



在该网段发现了Windows主机、linux主机、FreeBSD、MySQL、微软服务器、思科无线局域网控制器等

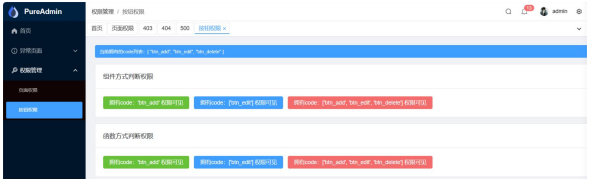
更便捷 查看设备信息



扫描工具拓展

- 发现很多设备开放80端口，尝试登录，千奇百怪
- 多数为后台管理系统，少数可用默认密码登录
- 扫描其他网络也有类似问题

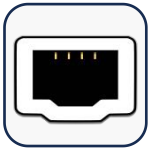
考虑网络安全问题！



扫描工具拓展

主动扫描 VS 被动扫描

主动扫描		被动扫描
主动发送数据包	技术方法	监听网络流量
更容易被目标主机或防火墙识别	可见性与隐蔽性	难以被察觉
通常信息更详细	准确性	准确性较低
增加网络流量或CPU负载	造成影响	不增加额外通信影响小



Crypto——加密算法

任务驱动的ICS技术学习

背景介绍

管理层

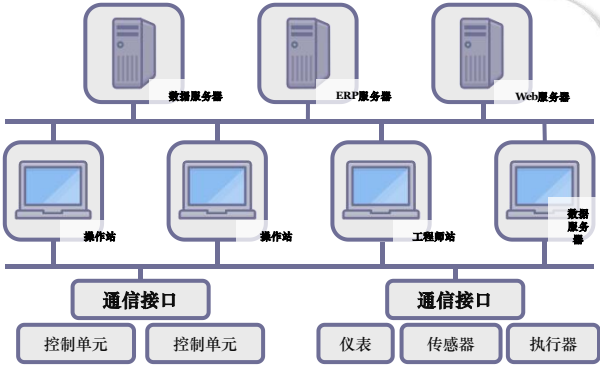
生产过程的计划和管理

监控层

任务部署、监视、调整

控制层

连接工业互联网与现场

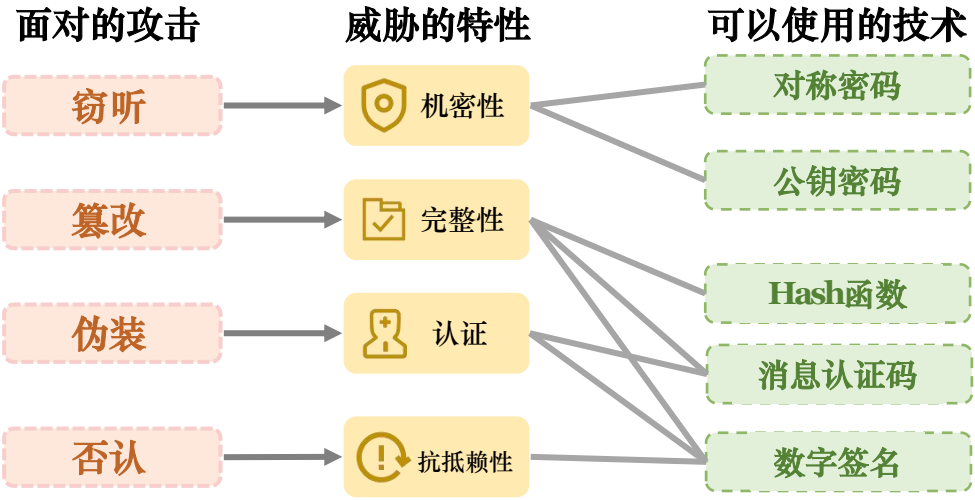


机密性 认证 完整性 抗抵赖性

自上而下分为管理层、监控层和控制层

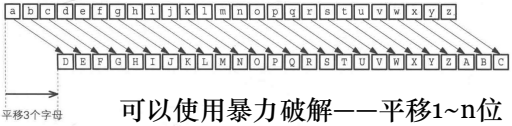
背景介绍

密码技术是核心问题



加密算法简述 —— 古典密码

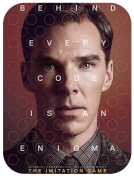
凯撒密码



可以使用暴力破解——平移1~n位

一道简单的Flag题目：MZC

Enigma加密



模仿游戏

替换密码

冷知识：英文文章出现最高的字母是e

字母替换，通过分析字频和词频可以破解

加密算法简述 —— 对称密码

一次性密码本

无法被破译(会出现所有序列)

DES | Data Encryption Standard

- 56-bit 对称密钥, 64-bit明文输入（每隔7bit会插入1bit错误检查的bit）
- 分组密码的一种
- 16 round Feistel网络

AES | Advanced Encryption Standard

- 数据128bit成组加密
- 穷尽法解密如果使用1秒钟破解DES,需要花149万亿年破解AES
- 通过竞争来实现标准化，彻底杜绝了隐蔽式安全性

Sub Bytes　Mix Columns　Shift Rows　Add RoundKey

Main Takeaway： 不要使用任何自制加密算法

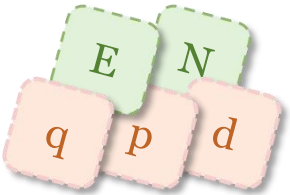
加密算法简述 —— 非对称密码

RSA | Rivest, Shamir, Adelson Algorithm

$$C = M^e \bmod n \qquad M = C^d \bmod n$$

公钥{E,N}　　私钥{D,N}

- ✓ 选择两个数 p 和 q。
- ✓ $n = pq$　n 是密钥长度。
- ✓ 计算$\varphi(n) = (p-1)(q-1)$
- ✓ 整数 e，使得 e 与 $\varphi(n)$互质
- ✓ e 的模逆 d使得$e \cdot d \equiv 1 \mod \varphi(n)$



以**大数质因数分解**
这个数学难题作为保障

对 RSA 的攻击

Lab Crypto Ch2

```
from sympy import mod_inverse
from Crypto.Util.number import long_to_bytes

p = 0x848cc7edca3d2feef44961881e358cbe924df5bc0f1e7178089ad6dc23fa1eec7b0f1a8c693
q = 0xa0ac7bcd3b1e826fdbd1ee907e592c163dea4a1a94eb03fd4d3ce58c2362100ec20d96ad858
e = 0x10001
c = 0x39f68bd43d1433e4fcbbe8fc0063661c97639324d63e67dedb6f4ed4501268571f128858b2f

# 因为给了p,q 相当于就给到了私钥了
n = p * q
phi_n = (p - 1) * (q - 1)

d = mod_inverse(e, phi_n) # 求模逆

pow(c, d, n) # 解密方法，求明文

m))
rney_is_very_fun_Phoenix_Wright&Miles_Edgeworth}
```

q p

Flag =AAA{Ace_Attorney_is_very_fun_Phoenix_Wright&Miles_Edgeworth}

ZJUAAA SimpleRSA

```
import gmpy2
from Crypto.Util.number import *

c=431396049519259356426983102577521801906916650819409770125821662319298730693

n=0x6270470b5e45bb464233683c30eeb03d17d54e0127036c9d286b00ac54946cfa1aa05c33e

e=3

def de(c, e, n):
    k = 0
    while True:
        m = c + n * k
        result, flag = gmpy2.iroot(m, e)
        if True == flag:
            return result
        k += 1

m = de(c, e, n)
print(m)
print(long_to_bytes(m))
```

E

Flag = b'AAA{touma_X1a03_qq_qun_386796080}'

加密算法——速度和安全性比较

DES

	10bits	1000bits	10^6 bits
Encryption Time /s	0.0005562305450439453	5.0067901611328125e-06	0.0010752677917480469
decryption Time /s	2.3126602172851562e-05	5.9604644775390625e-06	0.0010645389556884766

三重 DES

	10bits	1000bits	10^6 bits
Encryption Time /s	0.0008182525634765625	8.58306884765625e-06	0.0033576488494873047
decryption Time /s	2.9802322387695312e-05	8.106231689453125e-06	0.003479480743408203

速度较快
可被暴力破解

安全性提高
速度较慢

加密算法——速度和安全性比较

AES

	10bits	1000bits	10^6 bits
Encryption Time /s	0.0005285739898681641	3.5762786865234375e-06	0.00017762184143066406
decryption Time /s	2.217292785644531 2e-05	4.0531158447265625e-06	7.200241088671 88e-05

RSA

	10bits	1000bits	10^6 bits
Encryption Time /s	0.0035524368286132812	0.0005922317504882812	0.28232765197753906
decryption Time /s	0.002075672149658203	0.0020351409912109375	1.1794416904449463

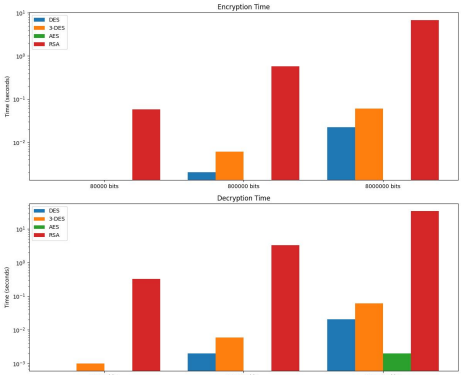
速度快
安全性高

Accept!

几乎无法暴力破解

二者为现行广泛采用的加密算法

加密算法——速度和安全性比较



名称	密钥长度	运算速度	安全性	资源消耗
DES	56位	较快	低	中
3DES	112位或168位	慢	中	高
AES	128、192、256位	快	高	低
RSA		慢	高 (取决于密钥长度)	高

加密算法——工控系统中各层需求及算法使用分析

管理层与监控层

算力较强	实时性要求低	主要采用AES-RSA 效率高、安全性高
核心中枢	实时性要求高	

控制层与监控层间、控制层内

算力资源有限	实时性要求高	主要采用 最快的AES算法

加密算法——工控系统中各层需求及算法使用分析

	国产密码算法		国际密码算法	
非对称加密	SM2	SM9	RSA	ECC
对称加密	SM1	ZUC	AES	DES
	SM7	SM4	3DES	
密码杂凑	SM3		MD5	SHA256

Thanks