Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

sense-of-security / ADRecon Public

Notifications Fork 281 Star 1.7k

Code Issues 17 Pull requests 2 Actions Projects Security Insights

Files

11881a2

Go to file

- Sample Output
- ADRecon.ps1
- LICENSE.md
- README.md

ADRecon / ADRecon.ps1

prashant3535 Typo 38e4aba · 6 years ago History

Code Blame 12056 lines (10805 loc) · 612 KB Raw

```
<#

.SYNOPSIS

    ADRecon is a tool which gathers information about the Active Directory and generate

.DESCRIPTION

    ADRecon is a tool which extracts and combines various artefacts (as highlighted bel
    The tool is useful to various classes of security professionals like auditors, DFIR
    It can be run from any workstation that is connected to the environment, even hosts
    Fine Grained Password Policy, LAPS and BitLocker may require Privileged user accoun
    The tool will use Microsoft Remote Server Administration Tools (RSAT) if available,
    The following information is gathered by the tool:
    - Forest;
    - Domain;
    - Trusts;
    - Sites;
    - Subnets;
    - Default and Fine Grained Password Policy (if implemented);
    - Domain Controllers, SMB versions, whether SMB Signing is supported and FSMO roles
    - Users and their attributes;
    - Service Principal Names (SPNs);
    - Groups and memberships;
    - Organizational Units (OUs);
    - Group Policy Object and gPLink details;
    - DNS Zones and Records;
    - Printers;
    - Computers and their attributes;
    - PasswordAttributes (Experimental);
    - LAPS passwords (if implemented);
    - BitLocker Recovery Keys (if implemented);
    - ACLs (DACLs and SACLs) for the Domain, OUs, Root Containers, GPO, Users, Computer
    - GPOReport (requires RSAT);
    - Kerberoast (not included in the default collection method); and
    - Domain accounts used for service accounts (requires privileged account and not in

    Author     : Prashant Mahajan
    Company    : https://www.senseofsecurity.com.au

.NOTES

    The following commands can be used to turn off ExecutionPolicy: (Requires Admin Pri

    PS > $ExecPolicy = Get-ExecutionPolicy
    PS > Set-ExecutionPolicy bypass
    PS > .\ADRecon.ps1
    PS > Set-ExecutionPolicy $ExecPolicy

    OR

    Start the PowerShell as follows:
    powershell.exe -ep bypass

    OR

    Already have a PowerShell open ?
```

```
        Already have a PowerShell open ?
        PS > $Env:PSExecutionPolicyPreference = 'Bypass'

        OR

        powershell.exe -nologo -executionpolicy bypass -noprofile -file ADRecon.ps1

    .PARAMETER Protocol
            Which protocol to use; ADWS (default) or LDAP

    .PARAMETER DomainController
            Domain Controller IP Address or Domain FQDN.

    .PARAMETER Credential
            Domain Credentials.

    .PARAMETER GenExcel
            Path for ADRecon output folder containing the CSV files to generate the ADRecon

    .PARAMETER OutputDir
            Path for ADRecon output folder to save the files and the ADRecon-Report.xlsx. (

    .PARAMETER Collect
        Which modules to run; Comma separated; e.g Forest,Domain (Default all except Kerber
        Valid values include: Forest, Domain, Trusts, Sites, Subnets, PasswordPolicy, FineG

    .PARAMETER OutputType
        Output Type; Comma seperated; e.g STDOUT,CSV,XML,JSON,HTML,Excel (Default STDOUT wi
        Valid values include: STDOUT, CSV, XML, JSON, HTML, Excel, All (excludes STDOUT).

    .PARAMETER DormantTimeSpan
        Timespan for Dormant accounts. (Default 90 days)

    .PARAMETER PassMaxAge
        Maximum machine account password age. (Default 30 days)

    .PARAMETER PageSize
        The PageSize to set for the LDAP searcher object.

    .PARAMETER Threads
        The number of threads to use during processing objects. (Default 10)

    .PARAMETER Log
        Create ADRecon Log using Start-Transcript

    .EXAMPLE

            .\ADRecon.ps1 -GenExcel C:\ADRecon-Report-<timestamp>
        [*] ADRecon <version> by Prashant Mahajan (@prashant3535) from Sense of Security.
        [*] Generating ADRecon-Report.xlsx
        [+] Excelsheet Saved to: C:\ADRecon-Report-<timestamp>\<domain>-ADRecon-Report.xlsx

    .EXAMPLE

            .\ADRecon.ps1 -DomainController <IP or FQDN> -Credential <domain\username>
        [*] ADRecon <version> by Prashant Mahajan (@prashant3535) from Sense of Security.
            [*] Running on <domain>\<hostname> - Member Workstation
        <snip>

        Example output from Domain Member with Alternate Credentials.

    .EXAMPLE
```

```
                Remove-Variable ADRObject
            }
            Remove-Variable ADRDomainAccountsusedforServiceLogon
        }

        $TotalTime = "{0:N2}" -f ((Get-DateDiff -Date1 (Get-Date) -Date2 $date).TotalMinute

        $AboutADRecon = Get-ADRAbout -Protocol $Protocol -date $date -ADReconVersion $ADRec

        If ( ($OutputType -Contains "CSV") -or ($OutputType -Contains "XML") -or ($OutputTy
        {
```

```
    {
        If ($AboutADRecon)
        {
            Export-ADR -ADRObj $AboutADRecon -ADROutputDir $ADROutputDir -OutputType $O
        }
        Write-Output "[*] Total Execution Time (mins): $($TotalTime)"
        Write-Output "[*] Output Directory: $ADROutputDir"
        $ADRSTDOUT = $false
    }

    Switch ($OutputType)
    {
        'STDOUT'
        {
            If ($ADRSTDOUT)
            {
                Write-Output "[*] Total Execution Time (mins): $($TotalTime)"
            }
        }
        'HTML'
        {
            Export-ADR -ADRObj $(New-Object PSObject) -ADROutputDir $ADROutputDir -Outp
        }
        'EXCEL'
        {
            Export-ADRExcel $ADROutputDir
        }
    }
    Remove-Variable TotalTime
    Remove-Variable AboutADRecon
    Set-Location $returndir
    Remove-Variable returndir

    If (($Protocol -eq 'ADWS') -and $UseAltCreds)
    {
        Remove-PSDrive ADR
    }

    If ($Protocol -eq 'LDAP')
    {
        $objDomain.Dispose()
        $objDomainRootDSE.Dispose()
    }

    If ($ADROutputDir)
    {
        Remove-EmptyADROutputDir $ADROutputDir $OutputType
    }

    Remove-Variable ADReconVersion
    Remove-Variable RanonComputer
}

If ($Log)
{
    Start-Transcript -Path "$(Get-Location)\ADRecon-Console-Log.txt"
}

Invoke-ADRecon -GenExcel $GenExcel -Protocol $Protocol -Collect $Collect -DomainControl

If ($Log)
{
    Stop-Transcript
}
```