

Why Cyber Crucible needs CRL/OCSP certificate access

Background

All Cyber Crucible communications from the endpoint agents to the CC infrastructure is protected by TLS. The list of domains used by CC agents, and each domain's function, is listed here:

[Which domains are used by Cyber Crucible?](#)

Access to those domains can be tested by an installer listed here:

[How can I test connection to these domains?](#)

Most of those domains listed are directly in control of by Cyber Crucible, and end in *.cybercrucible.com

A couple, notably the AWS Cognito domain required for some token and agent authorization capabilities, are not. Please note that, as of July 2025, agents no longer use Amazon Web Services (AWS) Cognito for machine-to-machine authentication.

By default, HTTPS libraries used by applications, including Cyber Crucible, verify the authenticity of TLS certificates by reaching out to the Certificate Authority (CA) servers via either CRL or OCSP.

This means that a firewall must not block the Certificate Authority domains either. These CA server domains typically service a large number of applications beyond Cyber Crucible, and can require additional justification from network security architect teams.

DMZ Mode

For environments in which the ability to adjust firewalls or other network security tools, to reduce or eliminate the number of tickets to the network security team, DMZ mode can be applied as a group setting.

DMZ mode enabled will do a few things:

- Insert the CA chains into any installer created for that group automatically.
- Automatically install the CA chain files into Cyber Crucible's folders.
- Automatically protect the CA chain files with Cyber Crucible's kernel level anti-tampering technology.
- Automatically configure the Cyber Crucible network communications to rely on the installed CA chain files for TLS certificate (HTTPS) verification.
- Automatically enable a service to ensure the CA Chains are maintained.

Enabling DMZ Mode for currently deployed agents

Turning on DMZ mode for a group with pre-deployed agents will attempt to task all existing agents to use the CA files, without user involvement.

This works as long as the agents are currently able to authenticate with CC infrastructure such as to receive this tasking. This means the agents currently do NOT need DMZ mode enabled, but will in the future. The likely scenario is either movement of the agents to a more restrictive firewall environment, or a desire to remove the existing firewall rules.

Enabling DMZ mode to CC agents which currently cannot authenticate due to CRL/OCSP errors in the CC service logs (likely found in C:\Program Files\CyberCrucible\service_log.0 or .1, etc), will not correct the matter. This can cause a chicken-or-the-egg scenario, if network security rules may not be adjusted at least temporarily, to allow the CC agents to receive the tasking to download and configure the CA chain files.

“Fire escape” method for enabling DMZ Mode for currently deployed agents

There are circumstances where Cyber Crucible agents are deployed without DMZ mode enabled, that cannot receive tasking, and which changing firewall rules to accommodate (even temporarily) to allow agents to receive DMZ tasking is difficult.

In those situations, please find the following digitally signed PowerShell script.

Three baseline requirements for this script to work:

1. This script must be run when the driver is disabled. This probably means Safe Mode, and NOT with Safe Mode Protection enabled in the CC portal. If it is enabled, please contact your Cyber Crucible support resources for assistance with a workaround.

2. This script must have the ability to contact "

<http://agent.tasking.rpp.cybercrucible.com/public/latestCARoots>"

3. This script is digitally signed, so altering the script will cause it to not run by default.

```
if ((driverquery /v | Select-String "CCRRSecMon").Matches.Length -gt 0) {
    Write-Error "Cyber Crucible is running."
    Write-Error "This script only works in safe mode with Cyber Crucible configured to not run i
    exit
}

$installPath = Get-ItemPropertyValue -ErrorAction SilentlyContinue -Path "Registry::HKEY_LOCAL_M
if ([string]::IsNullOrEmpty($installPath)) {
    Write-Error "Failed to get install path. Is Cyber Crucible installed?"
    exit
}
Write-Host "CC Install path: $installPath"

Write-Host "Fetching CA bundle..."
$json = (Invoke-WebRequest "http://agent.tasking.rpp.cybercrucible.com/public/latestCARoots").Co
if ([string]::IsNullOrEmpty($json)) {
    Write-Error "Failed to get CA roots."
    exit
}

Write-Host "Encoding CA bundle..."

$b64 = [System.Convert]::ToBase64String([System.Text.Encoding]::UTF8.GetBytes($json))
if ([string]::IsNullOrEmpty($b64)) {
    Write-Error "Encoding error."
    exit
}

$standardAlphabet = "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789+/"
$ccAlphabet = [byte[]](0x01,0x02,0x03,0x04,0x05,0x06,0x07,0x08,0x09,0x0a,0x0b,0x0c,0x0d,0x11,0x1
$b64 = ($b64.ToCharArray() | ForEach-Object {
    $index = $standardAlphabet.IndexOf($_)
    $ccAlphabet[$index]
})

$persist = New-Item -Force -Path $installPath -Name "persist" -ItemType "Directory"
if (-Not $persist.Exists) {
    Write-Error "Inner directory missing."
}

try {
    Set-Content -Value $b64 -Path "$installPath\persist\dmz.dat" -Encoding byte
    Write-Host "Updated CA bundle."
} catch {
    exit
}

try {
```

```
Set-ItemProperty -Path "Registry::HKEY_LOCAL_MACHINE\SOFTWARE\CyberCrucibleAgent" -Name "dmz"
Write-Host "Enabled DMZ mode."
} catch {
    exit
}
```

```
Write-Host -ForegroundColor Green "Success! Please disable safe mode and reboot."
```

```
# SIG # Begin signature block
# MIISPAYJKoZIhvcNAQcCoIISLTCCEikCAQExCzAJBgUrDgMCGGUAMGkGCisGAQQB
# gjcCAQSGwZBZMDQGCisGAQQBgjcCAR4wJgIDAQAABBAfzDtgWUsITrck0sYpfvNR
# AgEAAgEAAgEAAgEAAgEAMCEwCQYFKw4DAhoFAAQUhmRukJGB8aa0XQHFmv9zAjy6
# 2Ceggg6IMIIGsDCCBJigAwIBAgIQCK1ASmDSnEyfXs2pvZ0u2TANBgkqhkiG9w0B
# AQwFADBIMQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYD
# VQQLExB3d3cuZGlnaWNLcnQuY29tMSEwHwYDVQQDExEhEaWdpQ2VydCBUcnVzdGVk
# IFJvb3QgRzRwHhcnMjEwNDI5MDAwMDAwWhcnMzYwNDI4MjM1OTU5WjBpMQswCQYD
# VQQGEwJVUzEXMBUGA1UEChMORGlnaUNlcnQsIEluYy4xQTA/BgNVBAMTOERpZ2lD
# ZXJ0IFRydXN0ZWQgRzQgQ29kZSBTaWduaW5nIFJTQTQwOTYgU0hBMzg0IDIwMjEg
# Q0ExMIICiANBgkqhkiG9w0BAQEFAAOCAg8AMIICGKCAgEA1bQvQtAorXi3XdU5
# WRuxiEL1M4zrPYGXcMW7xIUUMJ+kjmjYXPXrNCQH4UtP03hD9BfXHtr50tVnGLJP
# DqFX/IiZwZHMgQM+TXAkZLON4gh9NH1MgFcSa00amfLF0x/y78tHWhOmTLMBICXz
# ENOLsvsI8IrgnQnAZaf6mIBJNYc9URnokCF4RS6hnyzhGMIazMXuk0lwQjKP+8bq
# HPNlaJGitiUyCEUhSan4QvRRXXegYE2XFf7JPhSxIpFaENdb5LpyqABXRN/4aBpTC
# fMjqGzLmysL0p6MDDnSlrzm2q2AS4+jWufcx4dyt5Big2MEjR0ezoQ9uo6ttmAaD
# G7dqZy3SvUQakhCBj7A7CdfHmzJawv9qYFSLScGT7eG0X0Bv6yb5jNWy+TgQ5ur0
# kfW+0/tvk2E0XLYTRSiDNipmKF+wc86LJiUGsoPUXPYVGUztYuBeM/Lo60wKp7AD
# K5GyNnm+960IHnWmZcy740hQ83eRGv7bUKJGyGFYmPV8AhY8gyit0Ybs1LCNU9D4
# R+Z1MI3sMJN2FKZbS110YU0/EpF23r9Yy3IQKUHW1cVtJnZoEUETWJrcJisB9I1N
# Wdt4z4FKPKBHX8mBUHOFECMHwWCKZFTBzCEa6DgZfGYczXg4RTCZT/9jT0y7qq0I
# U0F8WD1Hs/q27IwyCQLMbDwMVhECAwEAAa0CAVkwggFVMBIGA1UdEwEB/wQIMAYB
# Af8CAQAwHQYDVVR00BBYEFGg340u20/hfEYb7/mF7CIhL9E5CMB8GA1UdIwQYMBAA
# F0zX44LScV1kTN8uZz/nupiuHA9PMA4GA1UdDwEB/wQEAwIBhjATBgNVHSUEDDAK
# BggrBgEFBQcDAzB3BggrBgEFBQcBAQRrMGkwJAYIKwYBBQUHMAGGGGh0dHA6Ly9v
# Y3NwLmRwZ2ZlZXJ0LmNvbTB3BggrBgEFBQcwAoY1aHR0cDovL2NhY2VydHMuZGln
# aWNLcnQuY29tL0RwZ2ZlZXJ0VHJ1c3RlZFJvb3RHNCSjcnQwQwYDVROfBDww0jA4
# oDagNIYyaHR0cDovL2NybmDMuZGlnaWNLcnQuY29tL0RwZ2ZlZXJ0VHJ1c3RlZFJv
# b3RHNCSjcmwwHAYDVROgBBUwEzAHBgVngQwBAzAIBgZngQwBBAEwDQYJKoZIhvcN
# AQEMBQADggIBAD0jRD2NCHbuj7w6mdNW4AIapfhINPMstuZ0ZveUcrEAYq9sMCcT
# Ep6QRJ9L/Z6jfcBVN7w6XUhtldU/SfQnuxaBRVD9nL22heB2fjdxxyL3WqqQz/WT
# auPrINHVUHMImoqKwba9oUgYftzYgBoRGRjNYZmBVvbJ43bnx0QbX0P4PpT/djk9
# ntSZz0rdK0tfJqGVWEjVGv7XJz/9kNF2ht0csGBc8w2o7uCJob054Th02m67Np37
# 5SFTWsPK6WrXoj7bQ7gzyE84FJKZ9d30VG3ZXQIUH0AzfAPilbLCIXVzUstG2MQ0
# HKK1S43Nb3Y3LIU/Gs4m6Ri+kAewQ3+ViCCcPDMyu/9KTVCh4k4Vfc3iosJocsL
# 6TEa/y4ZXDlx4b6cpwoG1iZnt5LmTL/eeqxJzy6kdJKt2zyknIYf48FWGysj/4+1
# 6oh7cGvmoLr90j9FpsToFpFSi0HASIRLlk2rREDjjfAVKM7t8RhWByovEMQMCGQ8
# M4+uKIw8y4+ICw2/0/TOHnu077Xry7fwdxPm5yg/rBKupS8ibEH5glwVZsxsDsrf
# hsP2JjMMB0ug0wcCampAMEhLNKhRILutG4UI4lknBcoFUCvqShyepf2gpx8Gd0fy
# 1lKQ/a+fSCH5Vzu0nAPthkX0tGFuv2jiJmCG6sivqf6UHedjGzqGVnh0MIIH0DCC
# BbigAwIBAgIQBxoxP5I/FN11CdiDj62hWTANBgkqhkiG9w0BAQsFADBpMQswCQYD
# VQQGEwJVUzEXMBUGA1UEChMORGlnaUNlcnQsIEluYy4xQTA/BgNVBAMTOERpZ2lD
# ZXJ0IFRydXN0ZWQgRzQgQ29kZSBTaWduaW5nIFJTQTQwOTYgU0hBMzg0IDIwMjEg
# Q0ExMB4XDTE1MDEyMzAwMDAwMFoXDTE1MTAyOTIzNTk1OVowgdgxEzARBgsrBgEE
# AYI3PAIBAXMCVVMxGTAXBgsrBgEEAYI3PAIBAhMIRGVsYXdhcnUxHTABBgNVBA8M
```

FFBYaXZhdGUGt3JnYW5pemF0aW9uMRAwDgYDVQQFEwc1NjQ4MDE4MQswCQYDVQQG
EwJVUzEVMBMGA1UECBMMUGVubnN5bHZhbmhMRMwEQYDVQQHEwpQaXR0c2J1cmdo
MR0wGwYDVQQKExRDeWJlciBDcnVjaWJsZSwgSW5jLjEdMBsGA1UEAxMUQ3liZXIga
Q3JlY2libGUsIEluYy4wggiMA0GCSqGSIb3DQEBAQUAA4ICDwAwggIKAoICAQCc
gp0cezRZCGf8V3Uq625Qtx17DPHQGPSy0qj0J/BJlvKuaYkvMlh7tvzyakZm4e88
c10bk/99sz2Xz6+Gs00JmY32CYPVQ9crpliTHco1/Pde65kg0otHe9IvSQJAyrbz
J3PwnZB8tgFkWKkhaz7MSIjUmiZmrxDJ3Wzhga/87mH3RinBMY5d5C6TahBSyn5p
RGkzDokJ6zyTeh/XiI72RDH/SHGpfCgAfIz/dHuvZddp5YJ9QLhym5c65VfNHIZr
c8B9fvit6ADVXJbYfp1DR6C2WogfIJWxo5as+IuHm263qX96+ZoHR1jnTBegUCx6
z9yqIA1b5bwvWCRse9pFidWFjHVaEX/+eBKDeIeoFPsps+o3E4rGnrB4sDPNd6VZ
DqM9ddzkHNG5xWwxRIDx82G0VhyqGe8Ma26Je8ks95auuhB+Fpf7kqU80seX9Fqu
QaWqzz4PywddTmo3eigbdY0cwvYGF5zJh7aouzLEpZBXLvvAozXlpvLFTAU/LoCH
MkSnEHhY9wY9T40VKHxUG6YYYeydSeqdfV6fXrWfZwJbzsu1DKuUL92lNSlln+Zi
AFP1BB0hHdE3+p1nKztvGrJPCkRmZfb0TuQcl5PMMpmkjCQhebBjXVLbzka7kALt
QuXy9guLofly+WrlrLeAyC+mSy6KdAPlmn7jKYmGAwIDAQABo4ICAjCCAf4wHwYD
VR0jBBgwFoAUaDfg67Y7+F8Rhvv+YXsIiGX0TkIwHQYDVR00BBYEFMkijav6RdNE
3S+ZDCp309e0IGDdMD0GA1UdIAQ2MDQwMgYFZ4EMAQMwKTAnBggrBgEFBQcCARYb
aHR0cDovL3d3dy5kaWdpY2VydC5jb20vQ1BTMA4GA1UdDwEB/wQEAwIHgDATBgNV
HSUEDDAKBggrBgEFBQcDAZCBtQYDVR0fBIGtMIGqMF0gUaBPhk1odHRwOi8vY3Js
My5kaWdpY2VydC5jb20vRGlnaUNlcnRUCnVzdGVkRzRDb2RlU2lnbmLuZ1JTQTQw
OTZTSEezODQyMDIxQ0ExLmNybdBToFGgT4ZNaHR0cDovL2NybdQuZGlnaWNlcnQu
Y29tL0RpZ2lDZXJ0VHJ1c3RlZEc0Q29kZVNpZ25pbmdSU0E0MDk2U0hBMzg0MjAy
MUNBMS5jcmwwZGQCCsGAQUFBwEBBIGHMIGEMCQGCCsGAQUFBzABhhhodHRwOi8v
b2NzcC5kaWdpY2VydC5jb20wXAYIKwYBBQUHMAKGUGh0dHA6Ly9jYWNlcnRzLmRp
Z2ljZXJ0LmNvbS9EaWdpQ2VydFRydXN0ZWRHNEVZGVTaWduaW5nU1NBNDAA5NlNI
QTM4NDIwMjFDQTEuY3J0MAkGA1UdEwQCMAAwDQYJKoZIhvcNAQELBQADggIBAIO2
vz6Zr2BevD2bIH1qp/HoayEZmaDxFZbaJA0JGp2aa1o+I8jDFogpMw/sXg3tvU8i
l32/zDZjVj3g2AIfGnyVaFaVyQTUXMk+F/ODbsK9WogqV11ZopN05a01dvYKd5A8
+fH0gNDswEvUH6dqMrJuH11PU1hXDC2843Qb49SmBqKrY01ZvHML4IR6NDCPSa8
FyVapgwrYdvdYw5tA9r9eVkXzMXfSwT9yMUegZAW/Z6HSE2LPPyXWSUHPT0b8D8e
pWjmM8ruv4G3TbXZrgYIFBkVPQp9hd1MyUoYYPMTMUir3Zn/Ca7ak2rkE5Hch0+Q
+a0Ly4vS469ezh/PikriNwTy9zhSsCanF/T5In6X8LPCoN4m4Ex0LWEqtp5K+bD2
0ogz20BfHXTuIhujI0XMZqNwzyfE1iZ4S48G9Y8BBqk9KCY3P379CMBRlrsrsh0VE
OT98+zV3YEvDeAJrEDzvKzCBbeER086sGsPX6hRRtJ1x+XCbV+/2YJWW/y0/EsQ4
tKVWYw5DKQLZsJ66Co8qGKRY9RkHEps2BMrSaAopCbDkNnbVNi0PNZgMw6lXYbt2
yLG0hA/XNV6hBjSrku01hyXWu1Q0Let96IaGwYEIPMR/sHAUKi+JjIh/ObdnjboZ
DlvFCdMHPb7KvGVrcz0e600ACDVubDjn2jbE5T5aMYIDHjCCAxoCAQEwfTBpMQsw
CQYDVQQGEwJVUzEXMBUGA1UEChMORGlnaUNlcnQsIEluYy4xQTA/BgNVBAMTOERp
Z2lDZXJ0IFRydXN0ZWQgRzQgQ29kZSBTaWduaW5nIFJTQTQwOTYgU0hBMzg0IDlw
MjEgEQ0ExAhAHGjE/kj8U3XUJ2IOPraFZMAKGBSs0AwIaBQCgeDAYBgorBgEEAYI3
AgEMMQowCKACgAChAoAAMBKGCsGSIb3DQEJAzEMBgorBgEEAYI3AgEEMBwGCisG
AQQBgjcCAQsxDjAMBgorBgEEAYI3AgEVMCMGCSqGSIb3DQEJBDEWBWBBToaMqnekbd
9TNL+/F9FUVmT/wKpzANBqkqhkiG9w0BAQEFAASCAGCWUhyP7TV+jnFD0x3Ep+vL
/tn8ZW5LpurXf3/IzAA1GhiRy5Kv0D/3fgvV7Jv0/ASMwbm0Gqhnfkzj5A7G41JA
sZc18Bd+cz1bvUkd5ykUwhkQQ0t3Ha2yBsazwMGarZmPK6SE/NRp0pt2ZyPjr1A6
kJR0iX0rUDAUfAGLZNAhkhjtBu/x7Uq+8Q0aYCXlu1Nc01pkmHiELtMapq0Zx5k
pj+FYJT6tERBpp+79eE37eMpXU3paYJzHziq5vXS6aoaEQrxcvdMTXzz0ZS5w+AR
YAOoJCSHah0NsyeDdaQC0djLi2mF0mtleIlJATk6DyjC4N2Lu2ezXte3JH1Jqt1V
dDNcWmmzNKHAeSdX70ZjKSc44n9sMWA7wvLBxIYjqym3Qs5+VoMs910Di31WDMtF
80jS9zYxAmnJrMdV6stZQ4XLCN4/KQTCe7ioK7/PcTGNcBMNl5QIRsMWJ8dpqo0V
S7TNN+190u9QkDeTQ0c2C2ZVIMf43DNtkBFL2fvL+pZtH3IzMk+Ddpv1izJ4DV1y
p7qsxRLK9y4Idt+0KIorQLtMiGGDreeKZkNLKVz5is8YrjJ6kMwr1djhNeG+yLK2
FTZJhuybB22z7fRoC7Z1JBwBoGhiwvFSI+MJh9cbJN2X5lTXazxvK7QJSkcqwh

23476c/i0V1mHRRSCpvDKA==
SIG # End signature block

Revision #3

Created 2026-05-18 20:28:04 UTC by Dennis Underwood

Updated 2026-07-20 19:34:45 UTC by Dennis Underwood