

## Schubis Anleitung WinRM mit W2K8R2

### 1. winrm quickconfig

### 2. Check, ob überhaupt Kerberos enabled ist

```
winrm g winrm/config/service/auth
```

#### a. Einschalten von Kerberos für WinRM

```
winrm set winrm/config/service/auth @{Kerberos="true"}
```

bzw. bei PowerShell

```
winrm set winrm/config/service/auth ,{Kerberos="true"}
```

### 3. Prüfen ob WinRM funktioniert

```
winrm id -r:ifb-vc-02.ifbus.de -a:Kerberos -u:schubi-admin@ifbus.de
```

### 4. Zur Registrierung über http muss unverschlüsselter SOAP Transport erlaubt werden

```
winrm set winrm/config/service @{AllowUnencrypted="true"}
```

### 5. Das gleiche für den Client

```
winrm set winrm/config/client @{AllowUnencrypted="true"}
```

### 6. Normalerweise werden pro shell 1MB vorgesehen. Manchmal reicht das nicht und man muss den WinRM Processing Speicher erhöhen

```
winrm set winrm/config/winrs @{MaxMemoryPerShellMB="2048"}
```

### 7. krb5.conf Datei erstellen mit folgendem Inhalt (Großschreibung der Domains ist wichtig!!!)

```
[libdefaults]
    default_realm = IFBUS.DE
    udp_preference_limit = 1
[realms]
    ifbus.de = {
        kdc = IFB-DC-01.IFBUS.DE
        default_domain = IFBUS.DE
    }
[domain_realms]
    .ifbus.de=IFBUS.DE
    ifbus.de=IFBUS.DE
```

### 8. die Config Datei in das Java Security Verzeichnis kopieren

```
C:\Program Files\Common Files\VMware\VMware vCenter Server - Java Components\lib\security
```

Das muss als lokaler Admin erfolgen!!!

### 9. Der vCO Service muss neu gestartet werden

### 10. Prüfen, ob PowerShell Remote enabled ist

```
Enable-PSRemoting -Force
```

## 11. Im vCO den Workflow „Add a PowerShell host“ WF ausführen

- a. Als Hostname den FQDN des WinRM Servers angeben. IP funktioniert nicht mit Kerberos.
- b. Remote Host Type ist „WinRM“
- c. Transportprotokoll ist „http“
- d. Authentication ist „Kerberos“
- e. Session Mode ist „Shared“
- f. Nutzer muss ein Domain Nutzer mit lokalen Admin Rechten sein

## 12. Test des PowerShellPlugins mit „Invoke a PowerShell...“ WF

```
# Get set of adapters
$adapters = [System.Net.NetworkInformation.NetworkInterface]::GetAllNetworkInterfaces()
# For each adapter, print out DNS Server Addresses configured
foreach ($adapter in $adapters) {
    $AdapterProperties = $Adapter.GetIPProperties()
    $dnsServers = $AdapterProperties.DnsAddresses
    if ($dnsServers.Count -gt 0){
$adapter.Description
foreach ($IPAddress in $dnsServers) {
" DNS Servers ..... : {0}" -f $ipaddress.IPAddressToString
}
}
}
```

### Troubleshooting:

Wenn Remoteanmeldung mit „Error 400 http bad request (header too long“ erfolg, folgende Dinge beachten:

<http://support.microsoft.com/kb/970875>

<http://blogs.technet.com/b/shanecoathan/archive/2010/07/16/maxtokensize-and-kerberos-token-bloat.aspx>

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa\Kerberos\Parameters
Entry: MaxTokenSize
Data type: REG_DWORD
Value: 48000 (Decimal)
```

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\HTTP\Parameters
Entry: MaxFieldLength
Data type: REG_DWORD
Value: 65534 (Decimal)
```

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\HTTP\Parameters
Entry: MaxRequestBytes
Data type: REG_DWORD
Value: 1048576 (Decimal) -> that is the minimum value working in my case and means 1 MB;
maximum value is 16 MB (16777216 ) but try to keep it as low as possible
```