Ondřej Ševeček | PM Windows Server | GOPAS a.s. |
MCM: Directory Services | MVP: Enterprise Security |
ondrej@sevecek.com | www.sevecek.com |

WEB APPLICATIONS

Web Applications

WEB APPLICATIONS

Prerequisites

- Managed account for AppPool
- Database server and database name
- IIS site name, port, [host header], [ssl], [diskPath]
- AppPool name
- Authentication method
  - NTLM, Kerberos, Claims-based authentication
- Add-SPShellAdmin
- Define DNS CNAME/A
- Enable Kerberos SPN and AES

Requirements

- SP Admins must be DBCreator and SecurityAdmin in the DB instance
  - in case you are provisioning through Central Administration, the sp-farm must have the roles

Lab: Create Intranet...

- Create sp-intranet service account
  - add to SP Managed Accounts and Service Accounts groups
  - you can use the shadow group script
- In SQL instance DB1\SPContent create login
  - SP Admins, DBCreator + SecurityAdmin
- Define SharePoint managed service account
  - New-SPManagedAccount

Lab: ... Prepare to monitor OWSTIMER and WSSADMIN...

- On WFE1 start PROCMON
- Enable Filter to see only Path which begins with F:\SP-Intranet and/or applicationHost.config
- Start capturing events

Custom defaults for web.config

- %sharepointroot%\CONFIG
- web.config
  - the default web.config file which gets copied into each web application directory
- webconfig.something.xml
  - the <add> and <remove> elements to apply into the default web.config dynamically
- <configSections>
  - custom elements must be predefined unless are not read

Lab: ...Create Intranet

- New-SPWebApplication
  - -DatabaseServer spcnt
  - -DatabaseName SPIntranetContent
  - -Name SP-Intranet
  - -Path E:\SP-Intranet
  - -Url http://intranet
  - -Port 80
  - -AuthenticationMethod Kerberos
  - -ApplicationPool SP-Intranet-AppPool
  - -ApplicationPoolAccount gps\sp-web
- New-SPShellAdmin
  - -UserName "gps\sp admins"

Lab: Verify Intranet (WFE1)

- Verify DB exists and the SharePoint_Shell_Access DB role
- Verify IIS Web Site has been created
- Verify classic-mode Windows Authentication is enabled
  - Providers = Negotiate (Kerberos), NTLM
- Verify AppPool has been created and is started
  - verify the w3wp.exe process is not running yet
- Verify the E:\ disk path location has been populated with a basic set of files
  - web.config
  - check permissions contain WSS_WPG group

Lab: DNS and SPN

- Define DNS alias (A) for intranet
  - intranet = A = IP 10.10.0.55
- Define SPN and check uniqueness
  - setspn -a "http/intranet" sp-intranet
  - setspn -a "http/intranet.gopas.virtual" sp-intranet
  - setspn -q "http/*"
- AES for sp-intranet AD account
  - msDS-SupportedEncryptionTypes = 28

Lab: Create Extranet (SPCA)

- Create new managed service user sp-extranet
- New-SPAuthenticationProvider
  - -UseWindowsIntegratedAuthentication
  - -UseBasicAuthentication
  - -DisableKerberos $false
- New-SPWebApplication
  - -DatabaseServer spconfig
  - -DatabaseName SPExtranetContent
  - -Name SPExtranetInternal
  - -AppPoolName SPExtranetAppPool
  - -Url http://extranet
  - -AppPoolAccount gps\sp-extranet
  - - Port 80
  - -AuthenticationProvider $provider
- Add-SPShellAdmin

Lab: DNS and SPN

- Define DNS alias (A) for extranet
  - extranet = A = IP 10.10.0.57
- Define SPN and check uniqueness
  - setspn -a "http/extranet" sp-extranet
  - setspn -a "http/extranet.gopas.virtual" sp-extranet
  - setspn -q "http/*"
- AES for sp-extranet AD account
  - msDS-SupportedEncryptionTypes = 28

Lab: Create AppPool Profiles

- Log on to all SharePoint web servers and run Command Line as all of AppPool identities
- Verify that the local profile has been created in C:\Users folder

Web Applications

MANAGED PATHS AND SITE COLLECTIONS

Lab: Define Managed Paths

- New-SPManagedPath
  - -RelativeURL
  - -Explicit:$false
  - -WebApplication

Lab: Content Database

- New-SPContentDatabase
  - -Name
  - -WebApplication
  - -DatabaseServer
  - -MaxSiteCount
  - -WarningSiteCount
- Add-SPShellAdmin
- Test-SPContentDatabase
  - and verify on all WFEs that the binaries are ok

Lab: Site Collections

- Import-Module ActiveDirectory
- Get-Help Get-SP*template*
- New-SPSite
  - -Url
  - -ContentDatabase
  - -Description
  - -Language
  - -Name
  - -QuotaTemplate
  - -OwnerAlias
  - -SecondaryOwnerAlias
  - -Template blank

Lab: Sites

- New-SPWeb
  - -Url
  - -Description
  - -Language
  - -Name
  - -Template
  - -UniquePermissions

  - AddToTopNav, AddToQuickLaunch, UseParentTopNav

Lab: Test Kerberos Access

- Log on to Client1 as gps\sp-install
- Try accessing http://intranet/sales
- Use KLIST to display the http/intranet Kerberos ticket is present

Web Applications

REQUEST PROCESSING

Request Processing

- TCP session established
- HTTP.SYS in kernel mode
- WAS service starts W3WP.exe worker process for the AppPool
- W3WP authenticates the user if requested
- SPRequestModule produces output
  - from database or from disk
- ASP.NET compiles pages into memory or some to disk as well
  - %windir%\Microsoft.NET\Framework64\4.0\Temporary ASP.NET Files

Lab: Testing ASP.NET compilation

- Verify that 32bit framework does not run
  - %windir%\Microsoft.NET\Framework\4.0\Temporary ASP.NET Files
  - should be empty
- Delete contents of the 64bit framework cache
  - %windir%\Microsoft.NET\Framework64\4.0\Temporary ASP.NET Files
- Access http://intranet and watch the speed and the final size of the compilation results
  - verify that the folder does not contain anything like "home", "allitems" etc.

Web Applications

ALTERNATE ACCESS MAPPINGS

Extending vs. AAM

- Extend Web Application
  - up to 5 IIS web sites – zones
  - different authentication and user/anonymous policy
  - different port/host-headers on IIS bindings
- Alternate Access Mappings
  - is not meant as redirection means

Alternate Access Mappings

- Zones = URL sets
  - names are just labels
  - no functional difference except for Default
  - Default, Intranet, Extranet, Internet, Custom
- Public URL
  - what a users types into the address bar
  - all links and redirects get this URL
- Internal URL
  - what SharePoint accepts
  - should have equivalent host header binding

Zones

- URL sets
- Web Application User Policies
- Authentication

Lab: Extend intranet for SSL

- Use Central Administration to extend the http://intranet:80 web application
  - name SP-Intranet-SSL
  - path E:\SP-Intranet-SSL
  - port 443, use SSL
  - authentication Kerberos
  - zone Intranet

Lab: Changing IIS Binding

- Use Central Administration to delete all IIS sites and then extend the application to Default zone

Changing IIS bindings

- $wa = Get-SPWebApplication
- $site = $wa.IisSettings.Item('Default')
- $site.ServerBindings
- $bind = New-Object Microsoft.SharePoint.Administration.SPServerBinding
- $bind.HostHeader = 'intranet.gopas.virtual'
- $site.ServerBindings.Add($bind)
- $wa.Update()
- $wa.UnprovisionGlobally()
- $wa.ProvisionGlobally()

Lab: Change Classic to Claims

- Get-SPWebApplication
- Set-SPWebApplication
  - -AuthenticationProvider (New-SPAuthenticationProvider) -Zone Default
  - or
- .UseClaimsAuthentication = $true
- .Update()
- .MigrateUsers($true)
- .Deploy()

Lab: Change Claims back to Classic

- Get-SPWebApplication
- .UseClaimsAuthentication = $false
- .Update()
- .MigrateUsers($true)
- .Deploy()

Web Applications

CACHE

Caches

- No support in Foundation
  - Output and Object cache requires Publishing Feature
- Output cache
  - used only by Publishing feature
- Object cache
  - used only by Publishing feature
- Blob cache
  - local disk storage to cache files from SQL
  - separate from RBS storage on SQL server
  - does not require Publishing Feature

System Account

- System Account is moniker for AppPool account
  - do not log on to the site with AppPool account
- Some items are checked out and modified by System Account account
- Some items are created by the System Account

Portal Cache Users

- Portal Super User
  - reads all items including the drafts
- Portal Super Reader
  - reads only the published items
- Cache then checks if the calling user has permission to view the drafts and returns appropriate results
  - cache requests made only under two accounts and permissions validated "in application logic"

Default Portal Cache Users

- Portal Super User
  - by default System Account
  - performance problems
  - the checked out items must be downloaded twice
- Portal Super Reader
  - by default NT Authority\Local Service
  - does not work correctly with Claims Based Authentication
  - Access Denied for sites with SharePoint Server Publishing Infrastructure, Metadata Navigation or Content Query Web Part

Changing Portal Cache Users

- Define new AD accounts
  - define dedicated user accounts
  - the performance problem with checked-out items would be the same as with the System Account
- Define User Policy for both accounts
  - SuperAccount = Full Control
  - SuperReader = Full Read
- Assign the accounts to each application
  - use correct identity! (i:0#.w|gps\sp-intranet-cs)
- Enable Publishing Features for each site collection and all sites

Lab: Portal Users and Cache

- Create two AD user accounts
  - sp-superuser
  - sp-superreader
- Define application policies for http://intranet
  - sp-superuser – Full Control
  - sp-superreader – Full Read
- Assign portal cache users by usin PowerShell
  - $wa = Get-SPWebApplication
  - $wa.Properties['portalsuperuseraccount'] = 'i:0#.w|gps\sp-superuser'
  - $wa.Properties['portalsuperreaderaccount'] = 'i:0#.w|gps\sp-superreader'
  - $wa.Update()

Lab: BLOB Cache

- Open web.config for http://intranet web application
- Create E:\SP-Intranet-Cache
  - sp-intranet = Full Control
- Modify <BlobCache> element
  - enabled="true"
  - location="E:\SP-Intranet-Cache"

Web Applications

PERMISSIONS

Users

- Get-SPUser
- Move-SPUser

Access Control

- Permissions and inheritance
  - item/folder level
  - library/list level
  - site level
  - site collection administrators
- Groups/Users defined at site collection level
- Permission Levels
- User Policies at web applications
- User Policy Levels

Permission Inheritance

- SPWeb.HasUniquePerm
- SPWeb.BreakRoleInheritance
- SPWeb.RoleAssignments
  - list of permission entries
- SPWeb.Update()
  - must be called to update the settings into database

Groups

- SPWeb.SiteGroups
  - list of site collection defined groups
- SPGroup = SPWeb.SiteGroups.Add
- SPGroup.AllowMembersEditMembership
- SPGroup.Update()
  - must be called to upload changes to SharePoint

Users

- SPUser = SPWeb.EnsureUser(login)
- SPGroup.AddUser(SPUser)

Permission Levels

- SPWeb.RoleDefinitions["Contribute"]
- SPRoleAssignment = new-object Microsoft.SharePoint.SPRoleAssignment(SP Group)
- SPRoleAssignment.RoleDefinitionBindins.Add(SPRoleDefinition)
- SPWeb.RoleAssignments.Add(SPRoleAssignment)

User logins (Active Directory)

- UserInfo table in content database
  - individual data per site collection
  - never updated
  - bound with SID, even in case of Claims authentication
- (Get-SPFarm).MigrateUserAccount
- (Get-SPFarm).MigrateGroup
- Move-SPUser
- Set-SPUser -SyncFromAD
- stsadm.exe -o migrateuser
  -oldlogin <Domain name\Old user name>
  -newlogin <Domain name\New user name>
  -ignoresidhistory

User logins (FBA/LDAP)

- Bound by the LDAP search attribute value
- If AD LDS attribute changes
  - user must log in with the new AD LDS attribute
  - if allowed access by the new attribute value
  - new UserInfo provisioned

Web Applications

NETWORK LOAD BALANCING

Load Balancing Mechanisms

- DNS round robin
  - easy, cheap, no failover
- Network Load Balancing
  - easy, cheap, no devices required, may have problems with client state vs. excessive authentication traffic
  - precise load % distribution, restrict TCP/UDP ports
- Cookie-based HTTP proxy
  - expensive, the best from client state perspective
  - cookies are no security issue

DNS Round Robin

- Balancing
  - may be good to decrease TTL of the A records
- No failover
  - timeouts when a node is down
  - examples: ping, Word 2010 timeouts on first IP address unavailable
  - examples: IE7+, shared files, MSINFO32 can use the other IPs

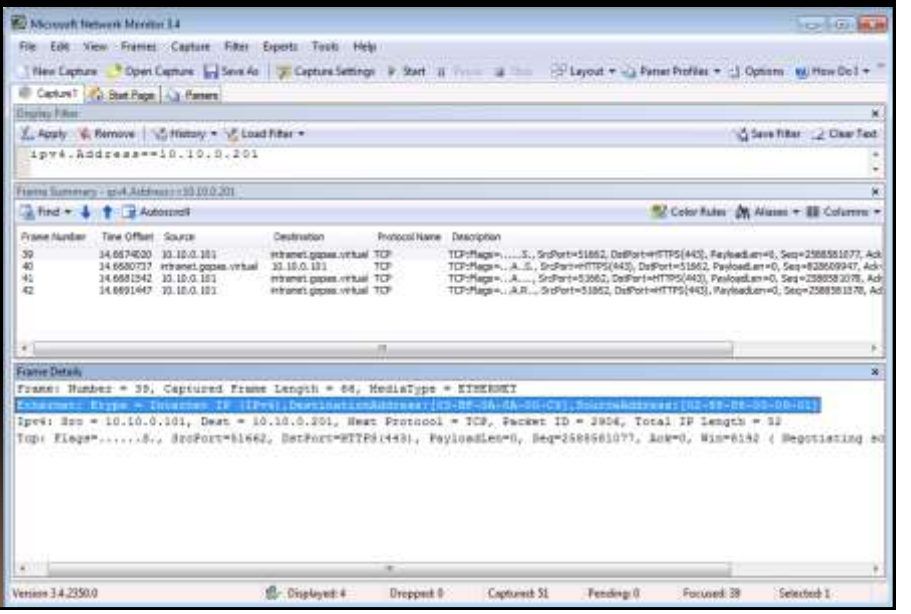
Network Load Balancing (NLB)

- No affinity (SHA-1 SourceIP:SourcePort)
  - may have problems with SP 2010 with AD FS
  - more authentication traffic with NTLM
  - problems with application state/cookies
- Single affinity (SHA-1 SourceIP)
  - not required with SP 2013 with AD FS
  - more stable state, not good for proxy clients
- C class affinity (SHA-1 SourceIP.IP.IP)
  - little use on intranets

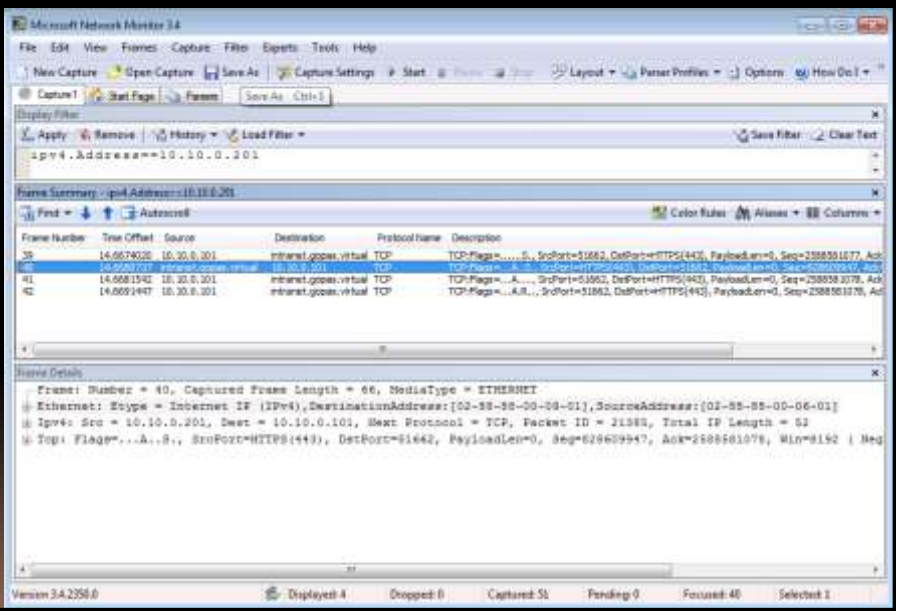
NLB Modes

- Unicast
  - unicast MAC address
  - the same MAC address on all nodes
  - no inter-node communications on the cluster NIC
  - must have inter-cluster NIC
- Multicast
  - multicast cluster MAC address
  - every node keeps its own real MAC address
  - no problem for inter-node communications
  - not supported by some switches
- Multicast with IGMP
  - nodes periodically send out IGMP to anounce their group membership
  - switch must have IGMP snooping enabled to limit flooding

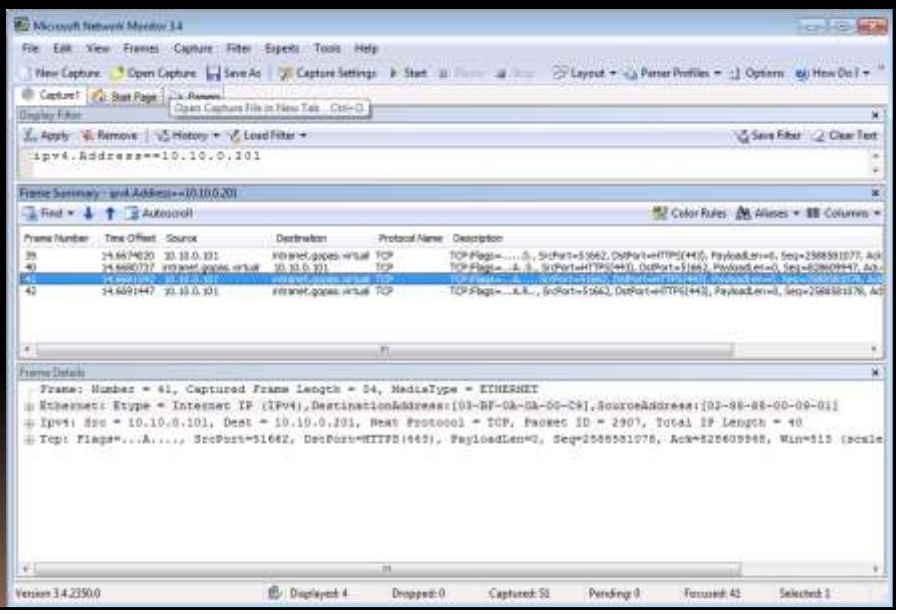
NLB Client - target cluster MAC



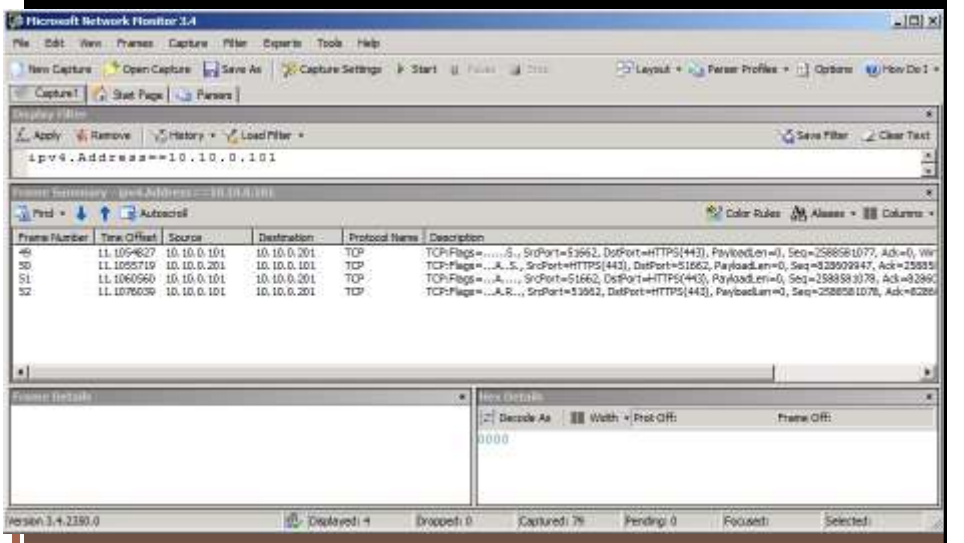
NLB Client - reply from real MAC



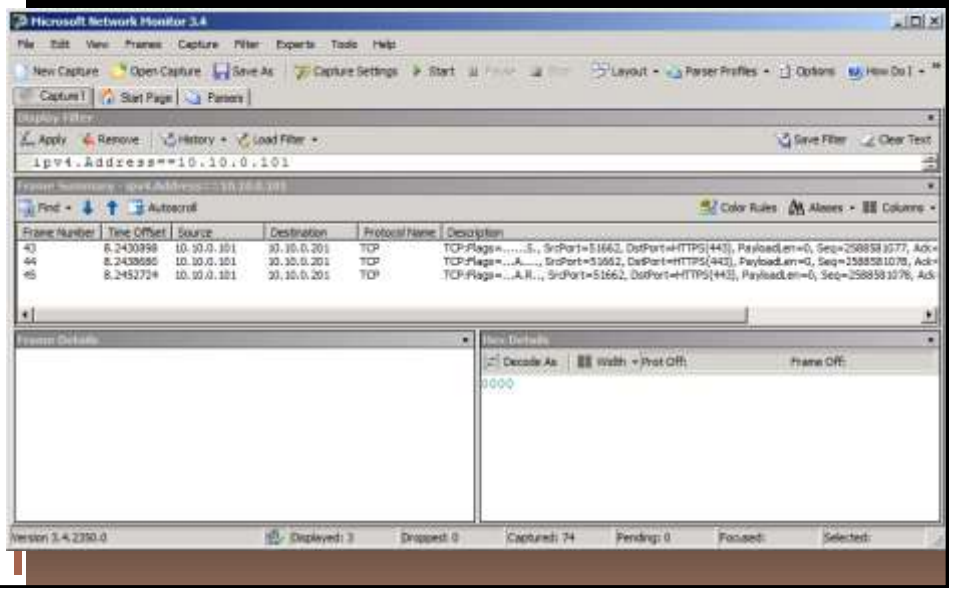
NLB Client - back to virtual MAC



NLB Node A - requests/reply



NLB Node B - requests only



Controlling the Host

- Stop/Start
  - stops handling all cluster traffic, drops all active connections
  - accepts all cluster commands
- Drainstop
  - stops handling new cluster traffic, keeps all active connections
  - after resume it goes to stopped state
- Suspend
  - the same as stop
  - after resume it goes to stopped state
  - does not accept any other cluster commands than resume and query

Scripting NLB

- NLB command line
  - NLBMGR is the GUI console
- WMI
  - root\MicrosoftNLB
  - SELECT * FROM MicrosoftNLB_Node
  - $node.Stop()
  - $node.Resume(); $node.Start()
  - $node.Suspend()

Cookie-based Proxy Balancing

- HTTP/S proxy issues its own cookie to the clients
- Maintains node affinity according to the cookie
- Office client applications started from browser will target the same node as well

Cookie Requirements for IE vs. Office Sharing

- Enable Persistent cookies
- Must reside in Trusted Sites zone or disable Protected mode for Intranet Sites
- Browser IE 8.0+ or the cookie must not contain HTTPOnly attribute (client-side scripts no access)
  - `Set-Cookie: jmeno=neco;expires=nekdy;path=/;domain=neco.gopas.cz;HttpOnly`

Web Applications

PERFORMANCE MONITORING

Processes

- W3WP.EXE
  - application pools, compiled in-memory contents
  - web applications, service applications
  - ideally use different user accounts and individual apppools for every service application
- OWSTIMER.EXE
  - jobs, a lot of work :-)
- LSASS.EXE
  - Windows authentication (Kerberos, NTLM)

Performance Monitor

- CPU: %
- CPU: time [hours:minutes]
- Memory: Private working set
- Memory: Pages/sec, Hard faults/sec
- Memory: Virtual memory or Commit charge
  - only in regard to memory leaks and paging file size
- Disk: Queue Length
- Disk: Response Time [ms]
- Disk: Read/Write Bytes/sec
- TCP: connections
- Network Interface: bandwidth?
- W3C_W3WP: Requests/sec
- W3C_W3WP: Active requests

Web Applications

LDAP AUTHENTICATION

LDAP "authentication" provider

- Not found in Foundation 2010/2013
  - LdapMembershipProvider
  - LdapRoleProvider