SYSTEMDOKUMENTATION HAJK LOGIN

**Ändringshistorik**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Datum | Ändringen avser | Utförd av |
| 0.9 | 2017-12-15 –  2018-02-27 | Skapad och uppdaterad | Victor Axelsson  Tobias Johansson |
| 0.91 | 2018-04-16 | Uppdaterad | Mattias Andrén  Hitomi Westrin |
| 1.0 | 2018-04-16 | Godkänd | Mattias Andrén  Hitomi Westrin |

**Innehåll**

1. Sammanfattning 3

2. Systembeskrivning 3

2.1. Kommunikation – Välja karta 3

2.2. Komponenter 4

2.3. Administrationsgränssnitt 4

2.4. Klient 12

2.5. Anrop 16

2.6. ArcGIS-server som provider 19

2.7. Felsökning 20

3. Installations- och konfigurationsbeskrivning 21

3.1 Kortfattad checklista för installation 21

3.1. Moduler och bibliotek 21

3.2. IIS 21

3.3. Konfiguration av Active Directory-kopplingen i backend 24

3.4. GeoServer 25

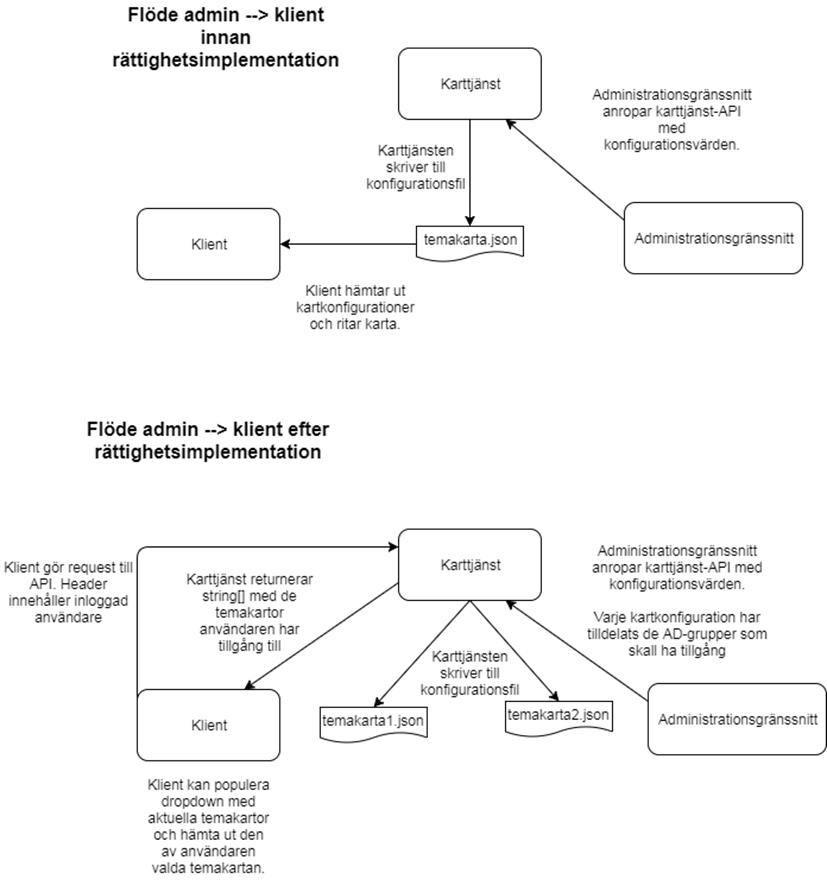
# Sammanfattning

Projektet ämnar utveckla stöd för inloggning och rättighetsadministration genom AD-grupper till kart-applikationen HAJK. Vidare utvecklas även möjlighet för slutanvändaren att välja olika tillgängliga temakartor direkt i klienten baserat på gruppindelning och rättighet.

# Systembeskrivning

## Kommunikation – Välja karta

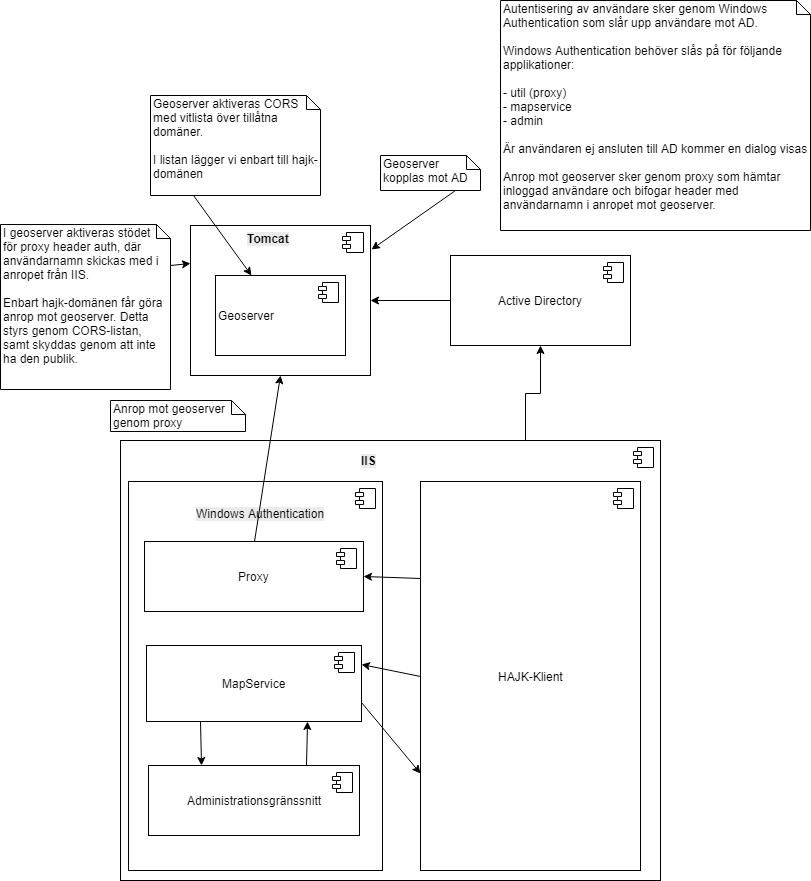
För att få en förståelse för hur de olika modulerna i HAJK kommunicerar med varandra, kan man referera till figur 1 som är en högnivåbeskrivning av kommunikationsvägarna



Figur .

## Komponenter

Figur 2 beskriver autentiserings- och auktoriseringslösningen.



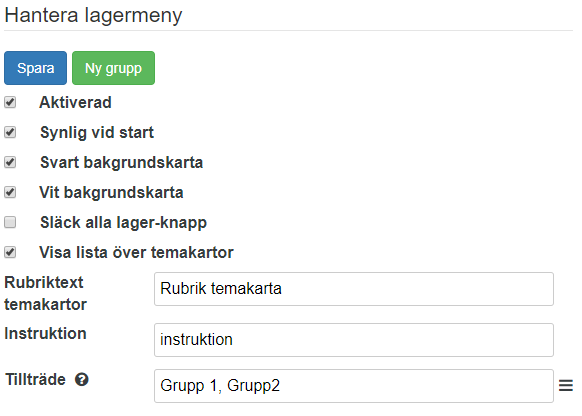
Figur 2.

## Administrationsgränssnitt

I administrationsgränssnittet finns nu en rad inmatningsfält som skriver till kartkonfigurationsfilen vid klick på spara. Alla fält som ingår i den nya funktionaliteten går att dölja genom att vid installation av hajk modifiera admin/config.json. Hajk letar efter en property vid namn "authentication\_active". Om denna är satt till true visas alla komponenter (inmatningsfält, checkboxar osv) som ingår, om den är satt till false eller inte finns, renderas de inte alls.

### Temakartor

Temakartor tilldelas på kartkonfigurationsnivå. När en ny karta skapas sätts de properties i kartkonfigurationsfilen som skall användas till tomma värden ([], ""). I gränssnittet ser inställningarna ut på följande vis:



För att ange vilka Active Directory-grupper som skall ha tillgång till en kartkonfiguration går man till fliken ”kartinställningar” följt av ”lagermeny”. Kryssrutan ”visa lista över temakartor” sätter en boolean som bestämmer om klienten skall rendera en lista över de temakartor inloggad användare har tillgång till eller ej. Inmatningsfältet bredvid rubriken ”Tillträde” används för att mata in en kommaseparerad lista med gruppnamn. Komma används som delimiter enligt standard RFC2253. För information om denna standard, se <https://www.ietf.org/rfc/rfc2253.txt>.

Whitespace är tillåten, men strippas från start och slut av varje sträng. Ex: ”Grupp 1” kommer att parsas med whitespace, men: ” Grupp 1” kommer att bli ”Grupp 1” efter parsning. Det är upp till administratör att ange grupper korrekt.

"Hamburgermenyn" till höger om inmatningsfältet "Tillträde" visar en lista med grupper som finns tillgängliga för administratören. Denna lista hämtas från Web.config i mapservice-tjänsten (parameter ”defaultADGroupsForAdmin”).

Gruppnamn för tillträde till en kartkonfiguration skrivs i konfigurationsfilerna till:



### Grupptilldelning verktyg

Alla verktyg som ingår i hajk har nu inmatningsfält som fungerar på samma sätt som i 2.3.1. Dessa har etiketten "Tillträde" och finns i alla verktyg så länge "authentication\_active" i config.json är satt till true.

Finns inga namn angivna för property “visibleForGroups” eller om den saknas så visas verktyget för alla grupper.

Gruppnamn för tillträde till ett verktyg skrivs i konfigurationsfilerna till:



### Grupptilldelning sökningar

Denna funktionalitet lägger till stöd för att tilldela tillträde till söktjänster (WFS:er) på kartkonfigurationsnivå. Detta sker genom att användaren kan se följande komponent i gränssnittet för sökverktyget (Kartinställningar => Verktyg => Sök):



Komponenten består av checkboxar för alla "wfslayers" som finns i layers.json. När användaren klickar i en checkbox visas inmatningsfältet för AD-grupper. Finns inga namn angivna alls för property "visibleForGroups" eller om den saknas så visas sökningen för alla grupper.

Dessa inställningar skrivs i konfigurationsfilerna till:



### Rollspecifika attribut vid infoklick

Verktyget "lagerhanteraren" utökas med möjligheten att skriva in vilka attribut som ska visas för ett lager i en specifik konfiguration. När en administratör klickar på ett lager som lagts till i lagerhanteraren under verktygsmenyn så finns möjligheten att fylla i en inforuta för att specificera vilka attribut som ska visas för lagret.

Den sträng som skall returneras vid infoklick för lagret anges i inmatningsfältet "Infobox". Denna box fungerar precis som inmatningsfältet för infobox när man i Hajk lägger till en visiningstjänst och preciserar hur infoboxen ska se ut.



Lagerspecifika attribut skrivs till konfigurationsfilerna till propertyn "infobox":



Om inget angivits i infobox eller om propertyn saknas, kommer den globala inställningen för visningstjänsten användas.

### Rollstyrning på lagernivå

Lagerhanteraren i administrationsgränssnittet utökas till att innefatta möjligheten att konfigurera roller som ska ha tillgång till respektive lager inom kartkonfigurationen. Administratören har möjlighet att klicka på de lager som har lagts till i verktyget lagerhanteraren i administrationsgränssnittet och specificera vilka grupper som ska ha tillgång till det specifika lagret i den aktuella kartkonfigurationen. Lagret kommer därmed inte visas i lagerhanteraren för de användare som har tillgång till kartkonfigurationen och vars grupp inte har tillgång till det aktuella lagret.

Detta innebär att en grupp som har tillgång till en kartkonfiguration inte behöver ha tillgång till alla de lager som finns tillgängliga i den aktuella kartkonfigurationen, pga rättighetsstyrning i GeoServer.

Samtidigt har inte ett lager i HAJK ett ett-till-ett förhållande till en tjänst i GeoServer då en administratör kan skapa upp ett lager i HAJK som består av flera lager från GeoServer. Dessa två fall måste administratören vara medveten om och se till att matchning av rättigheter blir korrekt för att inte lager ska vara klickbara i lagerhanteraren och låsta i GeoServer.

Tilldelning sker i administrationsgränssnittets lagerhanterare och ser ut på följande sätt:



Inmatningen "tillträde" tilldelar gruppnamn till det specifika lagret. Lager kan ha två olika states. De kan antingen var bakgrundslager eller ingå i en grupp.

Gruppnamn för tillträde till ett lager skrivs i konfigurationsfilerna till:

**Bakgrundslager**



**Lager som ingår i grupp**



Finns inga namn angivna alls för property “visibleForGroups” eller om den saknas så visas lagret för alla grupper.

## Klient

### Temakartor

Backenddelen i hajk har utöktas till att kunna hantera temakartor, till vilka åtkomst specificeras upp i respektive kartkonfiguration. I lagerhanteraren finns det nu möjlighet att lägga till en dropdownlista som håller möjliga kartkonfigurationer att byta mellan. Denna populeras med de kartkonfigurationer som uppfyller följande krav:

1. Kartkonfigurationen har nyckeln ”dropDownThemeMaps” i det objekt som har ”type” = ”layerSwitcher” under objektet ”tools” som ligger i roten.
2. Nyckeln ”dropDownThemeMaps” är satt till true.
3. Kartkonfigurationen har nyckeln ”title” under objektet ”map” i roten.
4. Kartkonfigurationen har nyckeln ”visibleForGroups” i det objekt som har ”type” = ”layerSwitcher” under objektet ”tools” som ligger i roten.
5. Den windowsanvändare som gör anropet till Hajks backend har tillgång genom ActiveDirectory, till de grupper som finns specificerade i arrayen ”visibleForGroups”. Finns inga namn angivna alls för property “visibleForGroups” eller om den saknas, så visas kartan för alla grupper.

Skulle något av dessa krav inte uppfyllas så visas inte kartkonfigurationen i den dropdownlista som finns i lagerhanteraren. Loggning sker även om krav 1, 2, 3 och 4 inte uppfylls.

### Rollspecifika sökningar

Delen av backenden som sköter inläsning av konfigurationer till klienten har utökats till att endast returnera WFS-lager (sökbara lager) beroende på vilka AD-grupper en specifik användare tillhör.

Vid ett klientanrop till backenden för att ladda konfiguration kontrolleras om det finns en inloggad användare eller ej. Finns det en inloggad användare så modifieras den inlästa konfigurationsfilen i minnet (filen skrivs ej om) till att endast returnera de lager från kartkonfigurationen som den inloggade användaren har tillgång till.

Varje kartkonfiguration som ska använda rollspecifika eller kartkonfigurationsspecifka söktjänster har fått en array "layer" på objektet "root"-"tools"-"search"-"options". Tillgången styrs av objektet "visibleForGroups" som finns specificerat för varje söklager i kartkonfigurationsfilen.

Klienten har även utökats med en override-funktion som använder de lokala id-numren som finns specificerade i de sökbara lagren i kartkonfigurationen istället för de globala söktjänsterna. Overriden sker endast om denna array finns tillgänglig på kartkonfigurationsobjektet.

### Rollspecifika attribut vid infoklick

Kartkonfigurationen utökas med nyckeln "infoBox" för varje lager i konfigurationen för "layerswitcher". Om administratören har specificerat attribut som ska visas på lagernivå i lagerhanterarens tillagda lager så kommer nyckeln populeras med dessa.

Nyckeln kommer läsas av vid start av HAJK och om nyckeln har ett värde kommer detta användas istället för de "default"-värden administratören har angett när lagret har lagts till som en valbar visningstjänst i Hajk.

Om en användare gör ett infoklick på ett lager som har laddats genom en av HAJK:s proxies så kommer denna strippas bort för att klienten inte ska skicka ett anrop genom en proxy som redan har skickats genom en proxy tidigare.

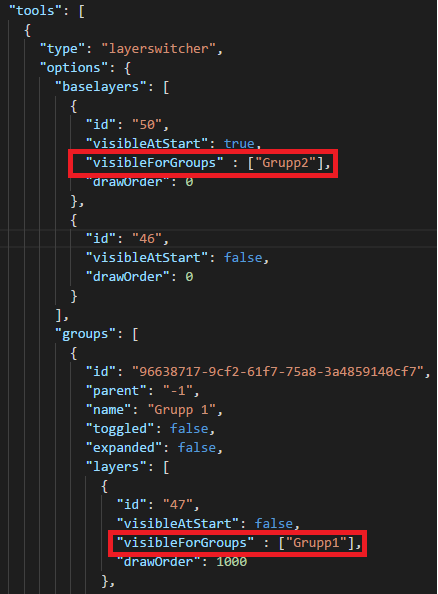
### Rollspecifika lager

Delen av backenden som sköter inläsning av konfigurationer till klienten har utökats till att endast returnera lager beroende på vilka AD-grupper en specifik användare tillhör.

Vid ett klientanrop till backenden för att ladda konfiguration kontrolleras om det finns en inloggad användare eller ej. Finns det en inloggad användare så modifieras den inlästa konfigurationsfilen i minnet (filen skrivs ej om) till att endast returnera de lager från kartkonfigurationen som den inloggade användaren har tillgång till.

Backenden läser in grupphierarkin "root"-"tools"-"layerswitcher"-"options" - "groups". Funktionen läser rekursivt genom alla grupper och modifierar objektet genom att ta bort de lager som den inloggade användaren inte får tillgång till. En användare får tillgång till de lager som har objektet "visibleForGroups" på lagret och innehåller en grupp som användaren är medlem i. Detsamma gäller för objektet "root" -"tools" -"layerswitcher" - "options" - "baselayers".

Finns inga namn angivna alls för property “visibleForGroups” eller om den saknas så visas kartan för alla grupper.



### Rollspecifika verktyg

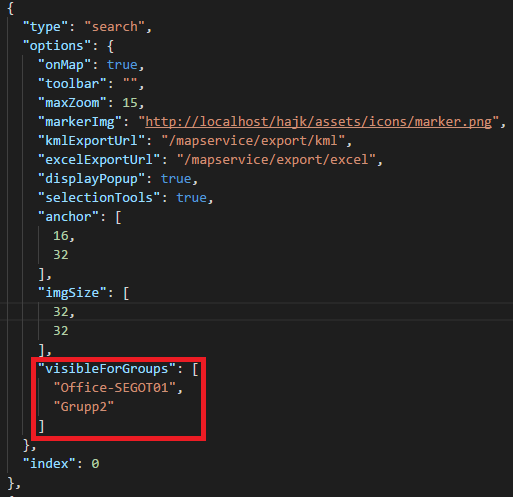
Delen av backenden som sköter inläsning av konfigurationer till klienten har utökats till att endast returnera verktyg beroende på vilka AD-grupper en specifik användare tillhör.

Vid ett klientanrop till backenden för att ladda konfiguration kontrolleras om det finns en inloggad användare eller ej. Finns det en inloggad användare så modifieras den inlästa konfigurationsfilen i minnet (filen skrivs ej om) till att endast returnera de verktyg från kartkonfigurationen som den inloggade användaren har tillgång till.

Backenden modfierar kartkonfigurationen genom att ta bort de verktyg i "root" - "tools" som den inloggade användaren inte har tillgång till. Funktionen letar efter arrayen "visibleForGroups" under respektive verktygs "options"-objekt.

En användare får tillgång till de verktyg som har objektet "visibleForGroups" på verktyget och innehåller en grupp som användaren är medlem i.

Finns inga namn angivna alls för property “visibleForGroups” eller om den saknas så visas kartan för alla grupper.



## Anrop

Klientkoden har korrigerats till att tvinga alla anrop genom proxyn även om de är interna. Detta för möjligheten att kunna styra rättigheterna och hämta inloggad användare när Windows Autentisering är aktiverad.

**För att utskrift av skyddade WMS:er och GetLegendInfo ska fungera måste fullständig sökväg till proxy preciseras i HAJK2.wmsproxy, inga relativa sökvägar. Man måste även ange rätt sökväg i Web.config för mapservice, se avsnitt 2.5.5.**

Proxysökvägarna ställs in i index.html på det globala objektet HAJK2. För att kunna använda skyddade tjänster krävs det att dessa specificeras annars kommer inte headern med användarnamn att skickas med till GeoServer. Det går som tidigare att köra utan proxy genom att inte specificera någon proxy på det globala objektet HAJK2.

### GeoServerAuthProxy

Implementationen av GeoServerAuthProxy bygger vidare på den befintliga GetProxyn som skickas med HAJK. De anrop som idag sker genom proxyn är endast domänöverskridande anrop men genom förändringar i klientkoden för HAJK tvingas även anrop som görs inom samma domän genom proxyn. Detta görs genom att på samma sätt som post-anrop dirigeras till postProxy, dirigeras nu alla get-anrop till den modifierade get-proxyn.

Proxyn hämtar ut användaren som gjort anropet genom ASP.NET impersonation. Om anropet innehåller en inloggad användare och om domänen som anropet skickas till finns med i nyckeln ”authorizedInternetDomains” i Web.Config för proxyn, så kommer en ny header läggas på requestet. Denna header kommer skickas med i anropet till GeoServer med det användarnamn som gjort anropet till proxyn.

GeoServer konfigureras därefter upp att läsa den specifika headern för att autentisera ett anrop. Namnet på headern kan ändras i Web.config för proxyn, default är namnet ”X-Control-Header”.

### Skyddad och icke-skyddad GeoServer

Tvånget genom proxy för anrop innebär att om Hajk använder samma proxy för att nå en skyddad GeoServer och en icke-skyddad GeoServer kommer de båda kräva Windows Autentisering. Skillnaden är att om URL:en som skickas till proxyn leder till en oskyddad tjänst kommer requestet inte att innehålla custom-headern. Proxyn går alltså att använda i en miljö där Windows Autentisering inte har aktiverats.



Figur . Schematisk bild där HAJK implementationen använder en Windowsautentiserad proxy.



Figur . Schematisk bild där HAJK implementationen INTE använder en Windowsautentiserad proxy.

### Vitlistning av anrop till GeoServer

Inloggningslösningen kräver att GeoServer litar på specifika domäner vid anrop så att inte vem som helst kan göra ett anrop med en korrekt header för autentisering. Detta kräver att domäner vitlistas och det kan göras på flera sätt.

Tomcat och Jetty är servlets som GeoServer körs i och genom att aktivera CORS på dessa och vitlista vissa domäner kan tillåtna anrop till GeoServer begränsas.

Ett annat alternativ är att sätta upp IP-blockeringar för de domäner som inte får anropa webservern.

I detta fall kommer detta skötas via IP-blockeringar genom att en context-fil överlagras i geoserver. Denna fil kommer få följande konfiguration:

<Context>

<Valve className="org.apache.catalina.valves.RemoteAddrValve"

allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:1" />

</Context>

Syftet med denna överlagring är att inga anrop mot geoserver kan ske förutom genom IIS, då det är den enda IP som är tillåten.

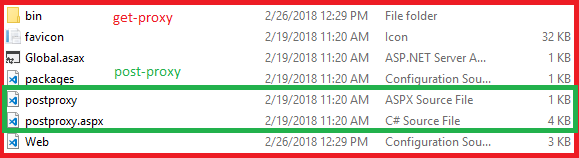
### Proxybeskrivningar

Det finns två proxyapplikationer i Hajk. En som hanterar get-anrop och en som hanterar post-anrop.

Get-proxyn finns i HAJKs git-repositiory under root/proxy/HTTPProxy och är en asp.net MVC-applikation. Denna skall, vid driftsättning i IIS motsvara applikationen 'util'.

Post-proxyn finns nu även den under root/proxy/HTTPProxy och består av två filer: postproxy.aspx samt postproxy.aspx.cs. Dessa två filer skall vid driftsättning placeras i samma mapp som Get-proxyn. Vid tidigare driftsättningar (Hajk 2.2 och tidigare) har denna funnits i projektets rootmapp. Nu skall den istället placeras i den mapp som håller get-proxyn. Detta för att 'util'-applikationen är under windowsautentisering och för att det skall gå att plocka ut inloggade användare. Något som inte är möjligt där filerna var placerad tidigare.

Mappen som håller båda proxy-lösningarna (och som motsvarar 'util'-applikationen i IIS) kommer alltså att se ut såhär:



**Get-proxy**

Följande sker kortfattat i get-proxyn:

* Stränghantering av eventuellt missformade protokollnamn
* Skapar ett WebRequest och hänger på:
  + CookieContainer.
  + Om användaren är påloggad (User.Identity.IsAuthenticated == true) och om anropet ska till en ”authorized” internet domän, sätts en header på requestet: X-Control-Header som innehåller påloggad användare.
* Anropet skickas iväg med GetResponseAsync().

**Post-proxy**

Är en ASP.NET proxy page. Denna hanterar följande:

* Plockar querystring-propertyn från requestet.
* Konfigurerar ett WebRequest och om användaren är påloggad och om anropet ska till en ”authorized” internet domän hängs X-Control-Header på requestet tillsammans med användarnamn.

Genom att sätta proxyadresser på det globala hajk-objektet så tvingar man alla anrop genom proxy-lösningarna. En möjlig väg runt detta är att låta proxyadresserna vara tomma och istället manuellt lägga på proxyadressen när man lägger till en visningstjänst i administrationsgränssnittet.

Syftet med båda proxylösningarna är att kunna göra anrop mot tjänster som inte finns på samma domän som klienten.

### Utskrift backend

I backendens konfigurationsfil finns en nyckel "ExportProxy" som styr utskrifterna av WMS:er genom proxyn. Denna måste sättas till en fullständig sökväg till GET-proxyn som används i HAJK för att utskrift av skyddade tjänster ska fungera. Detta krävs eftersom exportmodulen skapar upp ett WMS-lager på nytt för att kunna skriva ut det. Samtidigt krävs det att lagret som skickas in till Exportfunktionen går via proxyn redan i anropet till klienten för att visa lagret.

WFS-tjänsterna fungerar utan denna eftersom de laddas in i klienten och redan har autentiserats genom anropet till GeoServer och inte behöver skapas upp på nytt i exportmodulen.

## ArcGIS-server som provider

Systemet bygger på att GeoServer används som tjänst för att serva ut geografiska tjänster. Önskar man använda ArcGIS-server som provider finns det flera alternativ. Grundläggande krävs det att ytterligare en proxy sätts upp för att använda en annan provider och i fallet med ArcGIS-server kan man använda sig av exempelvis "esri resource proxy" eller ArcGIS-servers WebAdaptor.   
  
Proxylösningarna används för att autentisera användarna och sedan skicka en token vidare till ArcGIS-server för matchning mot roller som specificerats upp.

Nedan följer grova kodförändringar som måste göras för att ArcGIS-server ska stödjas som tjänst, detta måste givetvis utredas ytterligare.

1. GetLegendUrl i layercollection måste returnera en sökväg via den proxy som ArcGIS-server använder
2. ”url” i config för arcgis-lager måste utökas till att lägga på proxyadressen som ArcGIS-server använder vid anrop
3. GetFeatureInfo i arcgislayer.js måste skrivas om till att lägga på proxyadressen som ArcGIS-server använder på anropsurl.

## Felsökning

Loggning vid utskrift som beror på att en WMS-tjänst skickats in utan proxy går inte att fånga upp direkt där felet uppstår utan fångas i den allmänna catchsatsen, där kod har lagts till för att skriva ut errormessage till loggfilen.

Följande fel beror på att en proxy inte har skickats med till utskriften vid utskrift av en skyddad tjänst.   
  
{"Could not download capabilities document from the server. The server may not be available right now.Fjärrservern returnerade ett fel: (403) Förbjuden."}

# Installations- och konfigurationsbeskrivning

*Installera hajk enligt den generella systemdokumentationen men gör rättighetsinställningar enligt dokumentationen i avsnitten nedan*

## 3.1 Kortfattad checklista för installation

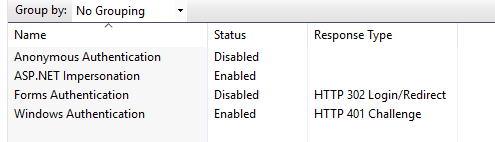
* Skapa upp site med Hajk med applikationerna admin, mapservice och util (util ska innehålla get- och postproxy).
* Siten Hajk ska ha Anonymous users = enabled.
* Admin, mapservice och util ska ha ASP.NET Impersonation och Windows Authentication = Enabled.
* Admin, mapservice och util ska också använda validateIntegratedModeConfiguration = ”false” under system.webServer i respektive Web.config:  
   <system.webServer>  
   <validation validateIntegratedModeConfiguration="false" />  
   </system.webServer>
* Ställ in absoluta sökvägar av proxy i index.html om dessa ska användas (krävs i framtiden för skyddade tjänster).
* Ställ in uppkopplingsparametrar för AD i Web.config i mapservice (backend).
* Ställ in proxysökväg i Web.config i mapservice för att i framtiden kunna använda utskrift av skyddade tjänster.
* Ge alla de användare som ska använda hajk rättighet att läsa och skriva till mapparna Temp och Upload och att läsa från App\_data. De användare som ska kunna uppdatera App\_data (via t ex Admin-gränssnittet) ska ges skriv-rättighet till denna mapp.
* Ställ in önskade proxysökvägar i admin/config.json och sätt "authentication\_active" till true om administrationsverktygen för AD-specifika inställningar ska användas.

## Moduler och bibliotek

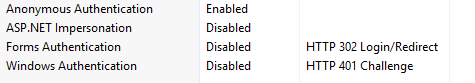
## IIS

För att de användarspecifika kartorna ska fungera måste IIS sättas upp till att autentisera användaren. Under HAJK-siten måste mapservice-applikationen och util konfigureras till att använda Windows Authentication och ASP.NET Impersonation. Detta görs genom följande steg:

* Gå till mapservice-/util-applikationen och välj sedan Authentication i ”Features view”.
* Ställ in Windows Authentication och ASP.NET impersonation till enabled.
* Ställ in Anonymous Authentication till disabled.



Huvudsiten Hajk som innehåller de olika applikationerna ska endast ha Anonymous Authentication = Enabled.



Gå tillbaka till ”Features View” och välj .NET Authorization Rules och lägg till en ”Deny”-regel för ”Anonymous Users”. Lägg sedan till de användare och grupper som ska ha tillgång till siten genom ”Allow rules”.

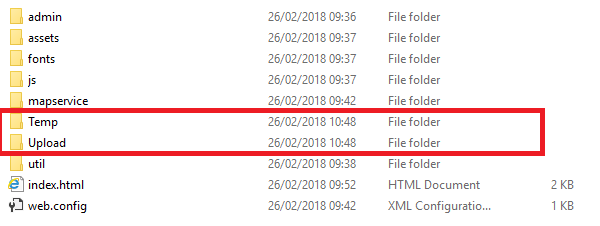
### Impersonation

IIS plockar ut, genom ASP-impersonation, den användare som har gjort anropet till mapservice. Användarens grupper hämtas sedan ut genom att göra ett uppslag mot ActiveDirectory med den aktiva användaren.

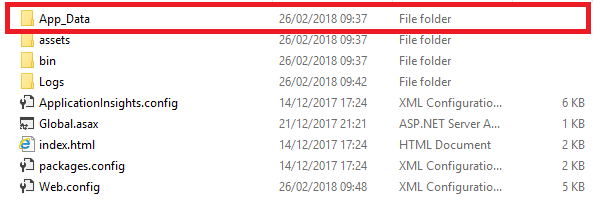
Detsamma sker för util-applikationen men här använder sig istället proxylösningarna av impersonation för att kunna skicka med den aktiva användaren i en header för att exempelvis användas i GeoServer för autentisering.

Hajk använder sig av ASP.NET Impersonation för att fastställa vilken användare som använder de olika applikationerna under siten Hajk. Detta innebär att alla de användare som kan tänkas använda Hajk, med rättighetsstyrning aktiverat, måste ha rättigheter att skriva till mapparna enligt bilden nedan:

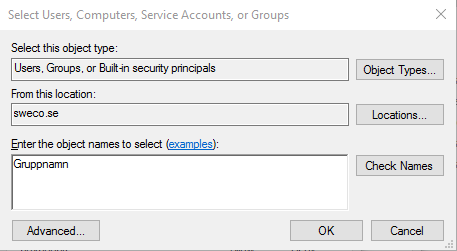
Mappen Temp och Upload som ligger i roten på Hajk-applikationen.

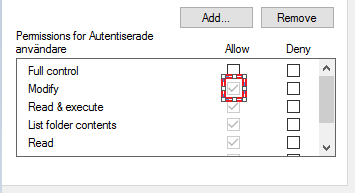


Mappen App\_Data som ligger under "hajkrot/mapservice"



Förslagsvis skapas en grupp med alla de användare som ska använda Hajk och sedan ges gruppen rättighet att skriva till de ovan nämna mapparna. Detta görs genom att gå in under "egenskaper" för mapparna och välja "säkerhet". Gör en editering och lägg till en användare/grupp med modifieringsmöjligheter för mappen.





## Konfiguration av Active Directory-kopplingen i backend

I web.config ligger fyra nycklar som används för att bestämma sökväg till domän och vilken ingångsnod som ska används för att söka efter användare. Två av nycklarna används även för att specificera vilken användare som kör frågorna mot Active Directory i koden.

|  |  |
| --- | --- |
| ActiveDirectoryContainer | LDAP-sökväg till den nod som sökningarna ska utgå ifrån i Active Directory.  Exempelvis  CN=Jeff Smith,CN=users,DC=fabrikam,DC=com |
| ActiveDirectoryDomain | Domännamnet |
| ActiveDirectoryUser | Användarnamnet som används för att koppla upp sig mot namnet på domänen |
| ActiveDirectoryPassword | Lösenordet för den användare som kopplar upp sig mot AD |
|  |  |

I web.config finns ytterligare en nyckel ”recursiveADsearch” vilken kan användas för att styra om sökningen på vilka grupper en specifik användare är medlem i ska ske rekursivt eller endast i de grupper som användaren är direkt medlem i. För ytterligare information se ”Remarks” <https://msdn.microsoft.com/en-us/library/bb335571(v=vs.110).aspx>.

## GeoServer

Tillståndshanteringen i GeoServer styrs genom att en specifik användare har ett antal roller. Regler sätts sedan upp på lager och tjänstenivå, till vilka roller knyts för att styra vem som har rätt att få svar på specifika anrop.

### Konfiguration av roller och användare

GeoServer kan sättas upp till att hämta roller från specifika noder i AD. Under "Användare, Grupper, Roller" i geoserver går det att lägga till en rolltjänst vilken kopplas upp till AD genom ett antal LDAP-inställningar och filter. GeoServer mappar de grupper som tjänsten hittar och skapar upp roller utifrån dessa genom att ge grupperna prefixet ROLE\_. Rollerna kan sedan användas i GeoServer för tillståndstilldelning till användare.

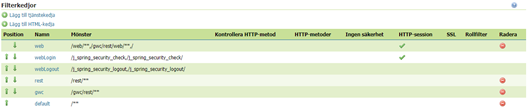
### Autentisering

GeoServer ska sättas upp för att göra autentisering genom ”http Header Proxy Authentication”. Detta görs genom att gå till ”Verifiering” och sedan skapa ett nytt Verifieringsfilter. Välj att skapa verifieringsfiltret ”HTTP Rubrik” och döp filtret till förslagsvis ”proxy”. Den rubrikrad som ska användas är ”X-Control-Header” (eller det namn på headern som används i Hajk) och innebär att GeoServer kommer slå upp headern på anropet och plocka ut värdet för att sedan matcha detta mot en användare i en vald användargruppstjänst. Sätt ”Rollkälla” till användargruppstjänst och välj den tjänst som har satts upp i GeoServer för att hantera användare.



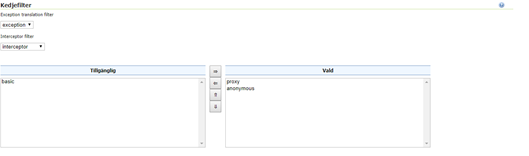
### Filterkedjor

GeoServer har under SäkerhetVerifiering ett antal filterkedjor som styr vad som ska hända om ett anrop sker mot en viss sökväg/ändpunkt. Genom att välja någon av filterkedjorna kan man styra anropet till att gå genom ett specifikt autentiseringsfilter.



Inställningarna här kan korrigeras utifrån behov men om man önskar att styra alla anrop till GeoServer genom det proxyautentiseringsfilter som skapats kan man välja att sätta detta på ”default”-filtret, vilket styr alla tjänster i GeoServer.

Välj ”default” i listan av filterkedjor och flytta sedan de olika verfieringsfiltren enligt bilden nedan. Detta betyder att vi avaktiverar basic-autentisering och väljer att proxyfiltret är det filter som först kommer att passeras för att försöka autentisera användaren.



### Restriktioner på tjänstenivå i GeoServer

I menyn under ”Säkerhet”  ”Tjänster” kan regler läggas till för de tjänster som GeoServer servar ut. Exempelvis kan en regel för WFS-ändpunkten läggas till och sedan knytas till ett visst antal roller för att styra vem som ska ha möjlighet att göra anrop mot WFS.

### Restriktioner på lagernivå i GeoServer

I menyn under ”Säkerhet”  ”Data” kan regler läggas till för att styra tillgången till separata lager. Detta styrs genom att en regel läggs till för ett valt lager och sedan knyts ett antal roller till regeln som administratören önskar ska ha tillgång till lagret. Här finns möjlighet att sätta upp regler för skriv-, läs- och administratörsrättigheter.