

RE: Confusion on standard security methodologies.

Source: <http://www.derkeiler.com/Newsgroups/microsoft.public.inetserver.iis.security/2003-01/7677.html>

From: Lisa Cozzens [MSFT] (lcozzens@online.microsoft.com)

Date: 01/16/03

From: lcozzens@online.microsoft.com (Lisa Cozzens [MSFT])

Date: Thu, 16 Jan 2003 01:02:23 GMT

> *I am getting a little confused on just how to set up a nice secure
> extranet.
>
> Here is the situation:
>
> Running an application that will require users to access the site both
> inhouse and remotely. Application will talk to a back-end SQL
> database.*

By "back-end," I assume you mean on a different box from IIS?

> *Here is my supposed configuration:
>
> -Application will use SQL NT authentication as a more secure method.
> Have the web server sit on the inside and open the necessary port
> (80/443) for remote clients to connect.*

If SQL is on a separate box, you won't be able to use NT authentication (also known as Integrated or Challenge/Response):

"Also, Windows NT Challenge/Response does not support double-hop impersonations (meaning that once passed to the IIS server, the same credentials cannot be passed to a back-end server for authentication, for example, when IIS uses Windows NT Challenge/Response, it cannot then authenticate the user against a SQL Server database on another computer by using SQL Integrated security)."

That's an excerpt from:

Q264921 INFO: How IIS Authenticates Browser Clients

<http://support.microsoft.com/default.aspx?scid=kb:EN-US:Q264921>

> *-How do remote clients authenticate to the domain? I would like to
> use Integrated Authentication with Kerberos, seems to be the standard
> - but may not be the best. I could do delegation with Basic Auth and
> Active directory. Or X.509 Certs mapped to accounts?*

You *can* use Kerberos authentication, but keep in mind that there are some very important restrictions on it:

1. The client must support Kerberos, so NT4, Win95, Win98, etc. won't be able to authenticate.
2. The client must be using Internet Explorer.
3. The KDC (key distribution center) must have port 88 open to the Internet for Kerberos traffic. Since your KDC is often a DC as well, your network admins may not be too happy about exposing it to the Internet.
4. The IIS server must be trusted for delegation. This is a security risk, since it means that if your IIS server is somehow compromised, an attacker will be able to get to other servers within your enterprise with much less difficulty.

Because of all these restrictions, Kerberos is rarely used across the Internet. Usually, when people need to authenticate users to a back-end server from IIS, they use Basic authentication, often with SSL to make it harder to sniff the password. Or, as Karl suggested, you can have people authenticate to IIS and then have IIS authenticate using a hard-coded username/password to SQL. You can either assume that if they've authenticated to IIS, they're OK to get into SQL, or you can pass over REMOTE_USER to SQL and manually do access checks based on that.

To be honest, I'm not sure how certificate mappings would work when you go out to the SQL server. Might work, might fail.

- > *–Some of these users do not have NT accounts, and really dont need to have it. I would like to keep this server off of my domain. Would I still need to create an active directory account for kerberos and then institute a domain trust.*

They're going to need to have the username/password of a valid NT account in order to authenticate to IIS. But this could be a local user (which wouldn't require any Active Directory, Kerberos, etc), and all the no-account users could share the username/password for this user if you want. The only potential problem with using a local user account is when you go to hit SQL. If you're using Basic so you can pass the credentials over to SQL, the SQL login will fail because the SQL server doesn't know anything about a local user account on the IIS server. This isn't an issue if you're doing a hard-coded username/password for all accounts to access SQL.

Another alternative would be to use a domain account as the no-account user. This would require the IIS server to be a member of a domain, but it could be in its own domain as long as you set up appropriate trusts between the IIS server's domain and your "real" domain. Note that you'll need to do this anyway if some of your users will be using domain accounts to authenticate.

- > *–Is it possible to use SQL mixed mode and for those that happen to have an NT account authenticate this way for others use a hash based authentication scheme?*

microsoft.public.inetsrvr.iis.security: RE: Confusion on standard security methodologies.

As I said above, NT authentication won't work if SQL is on a separate box from IIS.

I hope that helped clear some things up. It's impossible to say exactly how you should set up your extranet, because everyone has different requirements and priorities. For example, it might be fine in some situations if all users hit SQL using the same account, but that might not be desirable in other situations. It's up to you to weight the benefits and drawbacks of each scenario and decide what's best for you.

Lisa

Please do not send email directly to this alias. This is an online account name for newsgroup participation only.

This posting is provided "AS IS" with no warranties, and confers no rights. You assume all risk for your use.

© 2002 Microsoft Corporation. All rights reserved.