Eaglesoft 18 through 21 vulnerability

Update 02/01/2022: Eaglesoft 21.20 fixes this vulnerability [https://justinshafer.blogspot.com/2022/02/eaglesoft-2110-fixes-security.html] by requiring an attacker to know the Eaglesoft license being used. Great work.

What is Eaglesoft?

Eaglesoft is dental software that we call PMS or Practice Management Software. It holds the chart info, insurance, patient info, scheduling, scanned documents, and in some cases x-rays if the office is licensed for imaging.

Eaglesoft at one time relied on hard-coded credentials but has now changed the authentication.

When you install Eaglesoft with the server option, the installer installs Sybase SQL Anywhere with a username and password based off the license. It also creates a service called Patterson Application Service, and of course Eaglesoft client itself although this is an option for the server installation. You could just install the database and application service all by itself, though most people install all 3. The Patterson Application Service itself is a WCF Endpoint written in C#.

Client Authentication:

When you install Eaglesoft the client, the client doesn't know the credentials for the database. The client will talk to the Patterson Application Service over the LAN to get the credentials, but this is where the vulnerability is. To talk to the Patterson Application Service, you must use a certificate that is installed on the client and server version of Eaglesoft. The certificate itself is stored in the Windows certificate store. The certificate can be exported with the private key using the Windows certificate mmc console. First, the client will ask the Patterson Application Service for a list of Eaglesoft Users which is just a table in the database itself (not database users), to populate the main screen of Eaglesoft. At this point, the client still does not know the database credentials and is still talking to just the Patterson Application Service. If the password entered for the user is correct, then the Patterson Application Service will give the client the SQL Anywhere database credentials.

1. Eaglesoft runs, talks to Patterson App Service, gets a list of usernames for Eaglesoft.

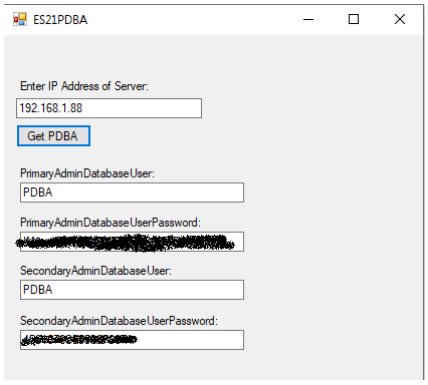
2. The end-user enters the password for an Eaglesoft User and if correct will then receive the database credentials for SQL Anywhere.

What is vulnerable?

This is a pretty good design, except that the Patterson Application Service isn't intelligent enough to know if someone has first gone through the Eaglesoft username and password authentication. The Patterson Application Service is trusting that the client has gone through this mechanism first (client validation). If someone reverse engineers the communications and learns the appropriate calls\methods, they could just write a program to ask the Patterson Application Service for the database credentials and the service will give them out to whoever is asking. The Patterson Application Service uses the certificate to encrypt the communications, so that is a requirement. The certificate is the same for all installations running on a specific version. Having certificates that are unique to each installation that is generated during the server installation would be good as well. Another problem is that the Patterson Application Service responds with base64 encrypted responses, but there is another call that allows you to ask the Patterson Application Server what the encryption keys are, and it too will just give them to you. The encryption key seems to be the same for all the installations I tested, although each installation has a different license. I like how Dentrix requires the workstation to know the Database Passphrase, which is just part of an algorithm that is converted to the actual database password, but the end-user doesn't know what that is. Or how Open Dental allows people to administer the database passwords themselves. If a workstation doesn't require anything unique to the installation\office, and can magically authenticate with Eaglesoft, then most likely there will be security problems.

Proof of Concept Tool

https://github.com/jshafer817/Eaglesoft [https://github.com/jshafer817/Eaglesoft]

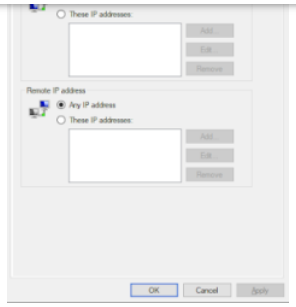


[https://1.bp.blogspot.com/-5Tpk4bIP3r8/YPn644Ay3HI/AAAAAAABmX4/ITEm_kVwhM0Val09CdqZbMHVvqssgtDWwCLcBGAsYHQ/s419/ES21PDBA.png]

One thing good about Eaglesoft is the SSN is encrypted in the database. They have a plain text column that shows the last 4 digits of the SSN and then a column that has the encrypted SSN. They are the only ones to do this to my knowledge and it has helped in one scenario I am aware of. A threat actor stole a database from an office I met on Facebook. The group was EGregor, and the office never wound up on their shaming site. I think a big part of that was the SSN was encrypted, but that is just a guess.

What can I do to mitigate this problem?

Firewalling the Patterson Application Server to restrict communication to trusted hosts is one good method to limit who can talk to the Patterson Application Server. One might be able to generate their own certificate and tell the Eaglesoft.Server.Configuration.data what thumbprint to use, as well as Eaglesoft.Client.Configuration.Data but I haven't tested this statement as it might also break the Patterson API Service.



[https://lh3.googleusercontent.com/-COHdUiqtCaQ/YPw3H7_BOCI/AAAAAAABmZQ/ozDrYxri3uoyOHOnYaGN7IS93dLH94X2gCLcBGAsYHQ/image.png]

How have I tested this?

I have access to about 11 different Eaglesoft installations and tested it on about half of them and all half I tested were vulnerable. The installations have different server names and licenses and all of them would give me the database credentials with a tool I put together. I used my laptop with a clean load of Windows to test, Eaglesoft was never installed on my computer. The version I tested is 21.00.18 which is the latest. This vulnerability also works if the Eaglesoft server is exposed to the internet.

US-CERT:

I contacted US-CERT about this issue. US-CERT is supposed to work between security researchers and vendors to coordinate vulnerabilities. According to US-CERT, Patterson Dental has not responded. US-CERT told me I should contact Mitre since they never responded, and that I should make my research public.

On Thu, Jun 11, 2020 at 9:30 AM CERT Coordination Center <cert@cert.org> wrote:

Greetings--

We will be closing this case on our end due to unresponsiveness from the vendor. We encourage you to request CVE ID(s) for your research in this case by visiting https://cveform.mitre.org/. Additionally, we encourage you to publish your research if you desire to do so; we have exhausted the avenues available for coordinated disclosure with the vendor's participation.

Thank you for your report, and please feel free to reply with any questions you may have.

[https://1.bp.blogspot.com/-pGw94FYWI5g/YRaCN8oUuMI/AAAAAAABmho/6_B7rjVjo5MkcqDwVUJw74sDlqC7O9RuQCLcBGAsYHQ/s1074/cert.png]

https://www.offthecusp.com/wp-content/uploads/2020/12/Eaglesoft_Security_Best_Practices.pdf
[https://www.offthecusp.com/wp-content/uploads/2020/12/Eaglesoft_Security_Best_Practices.pdf]

Posted 18th July 2021 by Justin Shafer

8 View comments

Music Lover July 18, 2021 at 12:32 PM
I still have no clue 🤦
Reply

Darrell July 19, 2021 at 6:15 AM
You are a good man, Justin. I think you may have a future in security!
Reply

Replies

Justin Shafer July 19, 2021 at 6:10 PM
Nope... this is my last stand...

Reply

Pavan Chakka August 16, 2021 at 6:16 PM
Hi Justin,

I have a question with regards to Eaglesoft SmartDoc. The documents imported through SmartDoc are stored in the data folder with .esd extension. I was wondering if you have any idea on what this file extension is?

We are working on a program (external) that automatically stores the patient documents in the Eaglesoft SmartDoc. We want the documents imported by this external program be read/available through SmartDoc as well. Inorder for the SmartDoc be able to read these files, we want to be able to store these files in the .esd extension/format as well. I was wondering if you have any idea on how to accomplish this.

I very much appreciate your help!

Thank you!
Reply

Replies

Justin Shafer August 16, 2021 at 8:33 PM
https://www.garykessler.net/library/file_sigs.html

Anonymous September 11, 2021 at 8:54 AM
From what I can tell it's just an encrypted ZIP archive. I don't know what the password used to encrypt these archives is, though. If you figure it out, Pavan, I'd greatly appreciate it!

Hi Justin - Thanks for all of the interesting reads on this stuff! I'm helping a dental group migrate off of Eaglesoft v18, and I'm wondering if it's possible to decrypt the SSNs so they can be migrated to new PM software (Denticon in our case). I heard anecdotally that Eaglesoft support will do this for you.. but Eaglesoft support isn't being very helpful :). I'm just wondering if anyone else has had success (or heard of) getting socials moved from Eaglesoft to new PM software.

Reply

Replies

Justin Shafer November 3, 2022 at 8:14 AM

I haven't tried that myself.

Reply

To leave a comment, click the button below to sign in with Google.

SIGN IN WITH GOOGLE

Load more