Talos Vulnerability Report

TALOS-2021-1301

CODESYS Development System ComponentModel ComponentManager.StartupCultureSettings Unsafe Deserialization vulnerability

JULY 26, 2021

CVE NUMBER

CVE-2021-21864

Summary

An unsafe deserialization vulnerability exists in the ComponentModel ComponentManager.StartupCultureSettings functionality of CODESYS GmbH CODESYS Development System 3.5.16 and 3.5.17. A specially crafted file can lead to arbitrary command execution. An attacker can provide a malicious file to trigger this vulnerability.

Tested Versions

CODESYS GmbH CODESYS Development System 3.5.16
CODESYS GmbH CODESYS Development System 3.5.17

Product URLs

https://store.codesys.com/codesys.html

CVSSv3 Score

7.8 - CVSS:3.0/AV:L/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H

CWE

CWE-502 - Deserialization of Untrusted Data

Details

The CODESYS Development System is the IEC 61131-3 programming tool for industrial control and automation technology, available in 32- and a 64-bit versions.

A file located in C:\Users<user>\AppData\Roaming\ or C:\ProgramData\ with the name APStartupCulture (but NOT APStartupCultre.xml) will cause the application to treat the file as a “legacy” file and pass it to BinaryFormatter.Deserialize().

The vulnerable method is _3S.CoDeSys.Core.Components.ComponentManager.StartupCultureSettings.method13()

```
private ComponentManager.StartupCultureSettings method_13()
{
    string text = this.method_16(ComponentManager.Enum0.const_2, false);
    if (string.IsNullOrEmpty(text))
    {
        text = this.method_16(ComponentManager.Enum0.const_2, true);
    }
    if (File.Exists(text))
    {
        Stream stream = null;
        ComponentManager.StartupCultureSettings result;
        try
        {
            if (Path.GetExtension(text) == string.Empty)
            {
                stream = File.OpenRead(text);
                result = (ComponentManager.StartupCultureSettings)new BinaryFormatter().Deserialize(stream); // [1]
            }
            else
            {
                result = new Class11().method_1(text);
            }
        }
        catch (Exception exception_)
        {
            Class3.smethod_1("Failed to load startup culture settings. Using defaults ", exception_);
            result = new ComponentManager.StartupCultureSettings();
        }
        finally
        {
            if (stream != null)
            {
                stream.Close();
            }
        }
        return result;
    }
    return new ComponentManager.StartupCultureSettings();
}
```

The `BinaryFormatter.Deserialize` method is never safe when used with untrusted input [2]. The deserialization that occurs at [1] is vulnerable to exploitation via an APStartupCulture file.

[2] https://docs.microsoft.com/en-us/dotnet/standard/serialization/binaryformatter-security-guide

Crash Information

Full Call Stack (application start with APStartupCulture file in C:\ProgramData\ and no APStartupCulture.xml file in C:\Users<user>\AppData\Roaming)

```
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.__BinaryParser.Run()
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.ObjectReader.Deserialize(System.Runtime.Remoting.Messaging.HeaderHandler handler
= null, System.Runtime.Serialization.Formatters.Binary.__BinaryParser serParser =
{System.Runtime.Serialization.Formatters.Binary.__BinaryParser}, bool fCheck = true, bool isCrossAppDomain = false,
System.Runtime.Remoting.Messaging.IMethodCallMessage methodCallMessage = null)
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(System.IO.Stream serializationStream =
{System.IO.FileStream}, System.Runtime.Remoting.Messaging.HeaderHandler handler = null, bool fCheck = true, bool isCrossAppDomain = false,
System.Runtime.Remoting.Messaging.IMethodCallMessage methodCallMessage = null)
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(System.IO.Stream serializationStream =
{System.IO.FileStream}, System.Runtime.Remoting.Messaging.HeaderHandler handler = null, bool fCheck = true,
System.Runtime.Remoting.Messaging.IMethodCallMessage methodCallMessage = null)
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(System.IO.Stream serializationStream =
{System.IO.FileStream}, System.Runtime.Remoting.Messaging.HeaderHandler handler = null, bool fCheck = true)
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(System.IO.Stream serializationStream =
{System.IO.FileStream}, System.Runtime.Remoting.Messaging.HeaderHandler handler = null)
mscorlib.dll!System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(System.IO.Stream serializationStream =
{System.IO.FileStream})
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager.method_13()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager.SpecificStartupCulture.get()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager.ComponentManager()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager2.ComponentManager2()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager3.ComponentManager3()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager4.ComponentManager4()
ComponentModel.dll!_3S.CoDeSys.Core.Components.ComponentManager5.ComponentManager5()
CODESYS.exe!_3S.CoDeSys.Main.CoDeSys.InitializeComponentManager(_3S.CoDeSys.Utilities.CommandLine cmdLine =
{_3S.CoDeSys.Utilities.CommandLine}, bool bNoUi = false)
CODESYS.exe!_3S.CoDeSys.Main.CoDeSys.<>c__DisplayClass16_0.<RunUI>b__0()
mscorlib.dll!System.Threading.ThreadHelper.ThreadStart_Context(object state = {System.Threading.ThreadHelper})
mscorlib.dll!System.Threading.ExecutionContext.RunInternal(System.Threading.ExecutionContext executionContext =
{System.Threading.ExecutionContext}, System.Threading.ContextCallback callback = {System.Threading.ContextCallback}, object state =
{System.Threading.ThreadHelper}, bool preserveSyncCtx = false)
mscorlib.dll!System.Threading.ExecutionContext.Run(System.Threading.ExecutionContext executionContext = {System.Threading.ExecutionContext},
System.Threading.ContextCallback callback = {System.Threading.ContextCallback}, object state = {System.Threading.ThreadHelper}, bool
preserveSyncCtx = false)
mscorlib.dll!System.Threading.ExecutionContext.Run(System.Threading.ExecutionContext executionContext = {System.Threading.ExecutionContext},
System.Threading.ContextCallback callback = {System.Threading.ContextCallback}, object state = {System.Threading.ThreadHelper})
mscorlib.dll!System.Threading.ThreadHelper.ThreadStart()
```

Serialization Exception

```
Failed to load startup culture settings. Using defaults (reason follows)
System.Runtime.Serialization.SerializationException: Binary stream '0' does not contain a valid BinaryHeader. Possible causes are invalid
stream or object version change between serialization and deserialization.
at System.Runtime.Serialization.Formatters.Binary.__BinaryParser.Run()
at System.Runtime.Serialization.Formatters.Binary.ObjectReader.Deserialize(HeaderHandler handler, __BinaryParser serParser, Boolean fCheck,
Boolean isCrossAppDomain, IMethodCallMessage methodCallMessage)
at System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(Stream serializationStream, HeaderHandler handler, Boolean
fCheck, Boolean isCrossAppDomain, IMethodCallMessage methodCallMessage)
at System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(Stream serializationStream, HeaderHandler handler, Boolean
fCheck, IMethodCallMessage methodCallMessage)
at System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(Stream serializationStream, HeaderHandler handler, Boolean
fCheck)
at System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(Stream serializationStream, HeaderHandler handler)
at System.Runtime.Serialization.Formatters.Binary.BinaryFormatter.Deserialize(Stream serializationStream)
at _3S.CoDeSys.Core.Components.ComponentManager.method_13()
```

Timeline

2021-05-18 - Vendor Disclosure

2021-07-26 - Public Release

CREDIT

Discovered by Patrick DeSantis of Cisco Talos.
