

## Protect Your Network with MFA

September 30, 2021

### History of the Password

In 1960, the Massachusetts Institute of Technology (MIT) developed a computer called Compatible Time-Sharing System (CTSS) that all researchers had access to. However, they shared a common mainframe as well as a single disk file. So, to help keep individual files private, the concept of a password was developed so that users could only access their own specific files for their allotted time.

After many decades, there have been several improvements to protect the password in rest, in use, and in transit using sophisticated encryption and hashing techniques. However, the concept itself has not changed. If your password is leaked or compromised, we have the same problem that we did on day one.

In 2019, an anonymous creator released 2.2 billion usernames and passwords freely across attacker forums, known at that time to be [the largest collection of breaches](#).

So, it's quite obvious that a single password has not stood the test of time.

### Beyond Passwords

We need something more than a single password...

1. In addition to the password (which will eventually be deprecated), you need to use something that you already have: a device (such as your cell phone).
2. The device needs to unlock using Face ID to ensure the best security.
3. In order to receive a one-time password (OTP) or a push notification, your device needs to have updated software and not be [jailbroken](#).

"By 2022, 60% of large and global enterprises and 90% of midsize enterprises (MSEs), will implement passwordless methods in more than 50% of use cases" -*Gartner Research*

One of the use cases we are going to talk about is protecting our network device login with Multi-Factor Authentication.

## Let's Talk Multi-Factor Authentication (MFA)

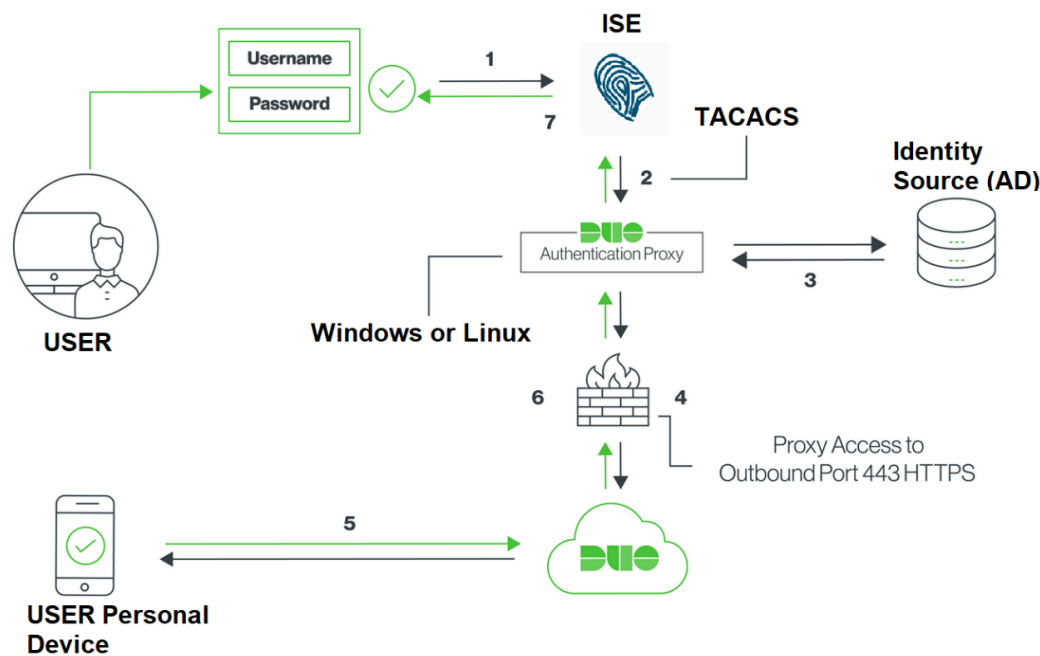
90% of customers that I encounter today still use their LOCAL, RADIUS, or TACACS enabled username and password to protect their critical network infrastructure. What we'll see in the remainder of this article is how easy it is to deploy MFA to your existing infrastructure to protect your network devices.

**This is where Cisco Duo does a great Job.** The application is not limited to only protecting network device logins; however, this is one area that I am often concerned about. Compromising one network device can lead to compromising the entire network. [Click here for a complete list of Duo capabilities.](#)

So, if you're interested in protecting your network device logins with MFA, then please continue reading!

## Setting Up MFA

### Flow of Events



1. Primary authentication initiated to ISE from user to access network device
2. ISE sends auth request to Duo auth proxy
3. Auth proxy server validates username/password from AD
4. Upon validation, auth proxy makes an API call to Duo security for second factor

5. Duo security sends push notification to end user's registered device
6. Device accepts Duo push notification; in turn, Duo responds back to auth proxy
7. Auth proxy informs send user validation to ISE; ISE assigned configured authorization profile and assigned Priv 15 level access in this case

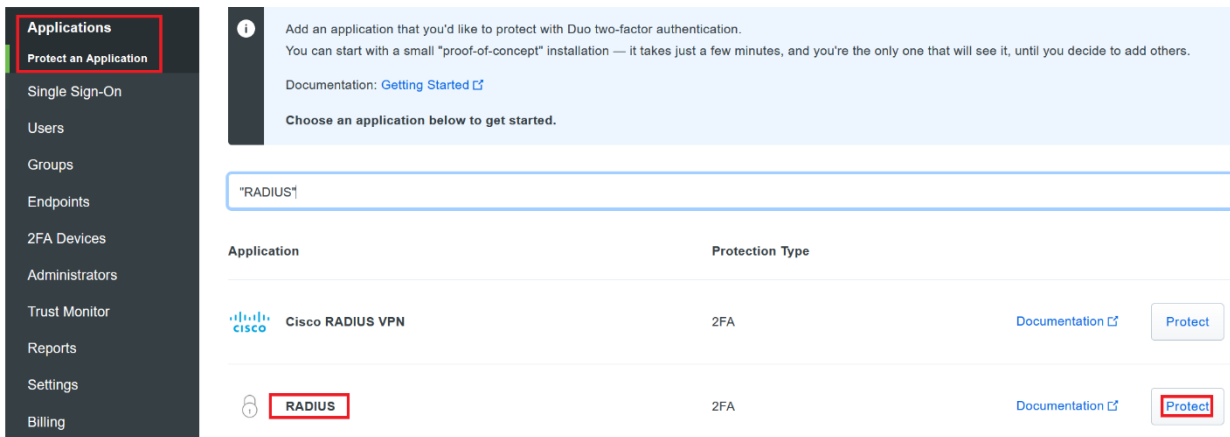
Now that we understand the basic flow of events, we clearly see there are **four key components** involved here:

1. Duo proxy server (in my case, installed on Win Server)
2. Radius/TACACS server and user identity database (in this case, I'm using ISE and Windows AD)
3. Network Infrastructure device we are trying to protect (in this case, Cisco 9800-CL controller)
4. End users who will use Duo for login



Our process will involve configuring these four components.

## 1. Duo Proxy Configuration

Start by **creating a free Duo account**, logging in, and clicking on the application you want to protect for TACACS login protection. I will select RADIUS.



The screenshot shows the Duo Admin Console interface. On the left, a navigation menu has 'Applications' highlighted with a red box. Below it, 'Protect an Application' is also highlighted. The main content area has a light blue header with an information icon and text: 'Add an application that you'd like to protect with Duo two-factor authentication. You can start with a small "proof-of-concept" installation — it takes just a few minutes, and you're the only one that will see it, until you decide to add others. Documentation: [Getting Started](#) Choose an application below to get started.' Below this is a search bar containing 'RADIUS'. A table with two columns, 'Application' and 'Protection Type', lists two items. The first item is 'Cisco RADIUS VPN' with a protection type of '2FA'. The second item is 'RADIUS' with a protection type of '2FA'. The 'RADIUS' application name and its 'Protect' button are highlighted with red boxes.

Application	Protection Type
 Cisco RADIUS VPN	2FA
 RADIUS	2FA

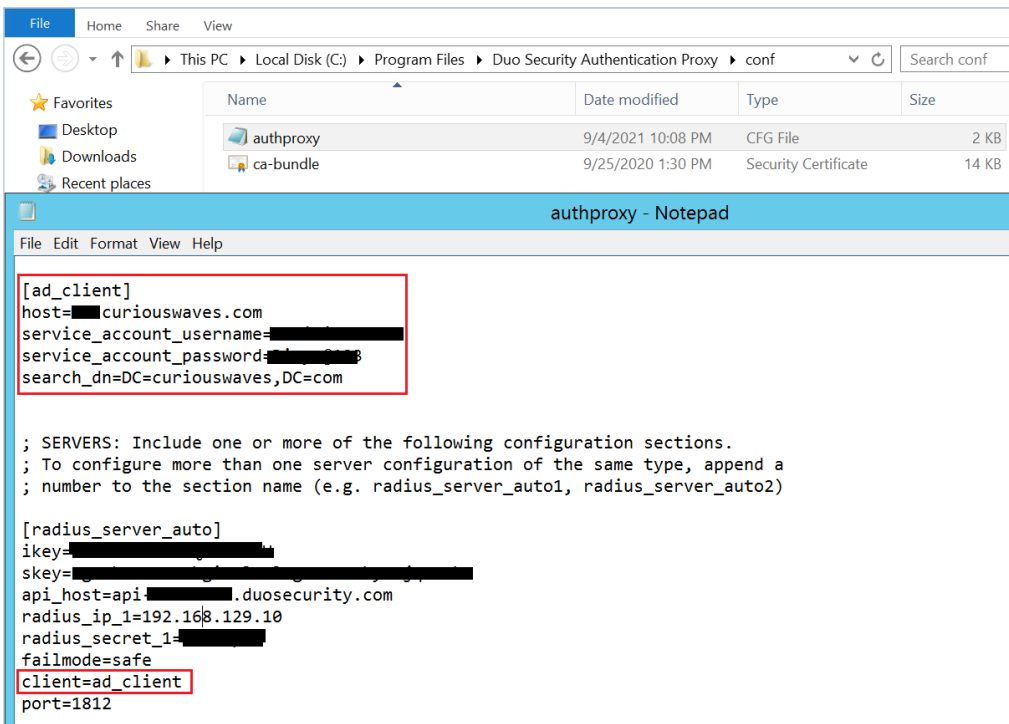
This will generate three keys:

- Integration Key: Secure API call between auth proxy server and Duo security
- Shared Secret: Secure trust between auth proxy and Duo cloud app
- API hostname: API call destination unique for the application

Make note of these keys in a secure location because we will be using them soon. These credentials should never be stored or transmitted in unsecure systems such as email, internal documentation / wiki pages, source code repositories, etc. They should only exist on the system(s) being protected by Duo.

Now we can **download and install Duo proxy**, which can be supported on variety of endpoints. In my case, I'm using Win Server 2012 (a lot of old stuff in my home lab, but it does the job!). [Click here](#) for more details on supported devices and how to install Duo proxy.

Once proxy is installed, configure it to be the bridge between your network and the Duo server by **configuring the AuthProxy file**. This is where we will use those three keys we talked about.

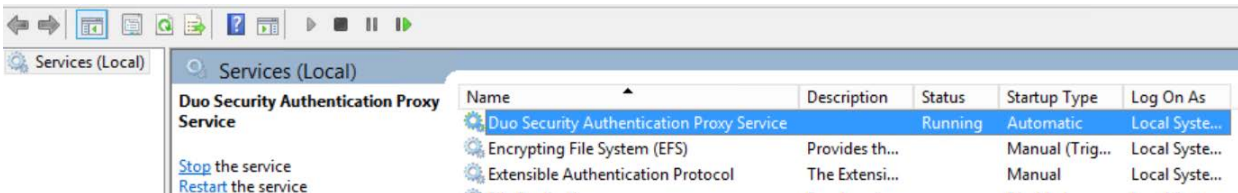


```
[ad_client]
host=curiouswaves.com
service_account_username=
service_account_password=
search_dn=DC=curiouswaves,DC=com

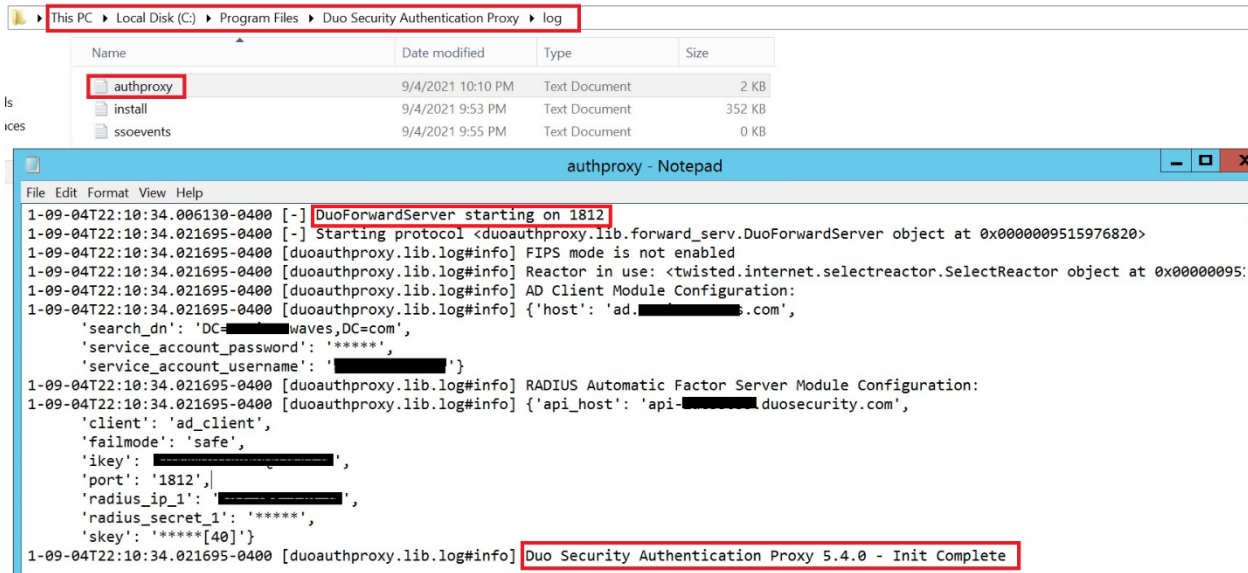
; SERVERS: Include one or more of the following configuration sections.
; To configure more than one server configuration of the same type, append a
; number to the section name (e.g. radius_server_auto1, radius_server_auto2)

[radius_server_auto]
ikey=
skey=
api_host=api. duosecurity.com
radius_ip_1=192.168.129.10
radius_secret_1=
failmode=safe
client=ad_client
port=1812
```

## Start the Duo Authentication Proxy Service and Check Logs to Ensure Connectivity



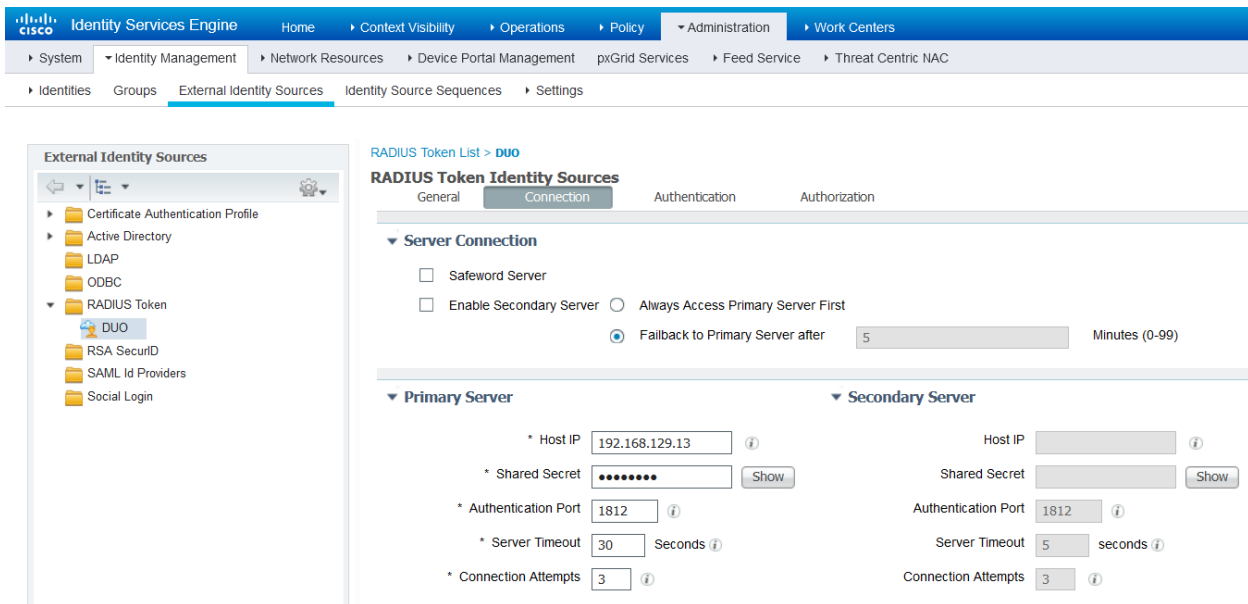
Name	Description	Status	Startup Type	Log On As
Duo Security Authentication Proxy Service		Running	Automatic	Local System...
Encrypting File System (EFS)	Provides th...		Manual (Trig...	Local Syste...
Extensible Authentication Protocol	The Extensi...		Manual	Local Syste...



Let's now configure our TACACS Server (ISE) to send request to Duo proxy server.

## 2. Configure ISE and User Identity

We will start by creating a new radius token named Duo (can be any name) with assigned Duo proxy server IP and shared secret (not the same as Secret Key used between Auth Proxy and Duo app). If you prefer, you can configure multiple servers as primary and backup.



## Create an Identity Source Sequence

Identity Services Engine Home Context Visibility Operations Policy Administration Work Centers

System Identity Management Network Resources Device Portal Management pxGrid Services Feed Service Threat Centric NAC

Identities Groups External Identity Sources Identity Source Sequences Settings

Identity Source Sequences List > DUO\_AD\_AUTH

### Identity Source Sequence

**Identity Source Sequence**

\* Name:

Description:

---

**Certificate Based Authentication**

Select Certificate Authentication Profile

---

**Authentication Search List**

A set of identity sources that will be accessed in sequence until first authentication succeeds

Available Selected

Internal Endpoints	<input type="button" value="&gt;"/> <input type="button" value="&lt;"/> <input type="button" value="&gt;&gt;"/> <input type="button" value="&lt;&lt;"/>	DUO	<input type="button" value="&lt;"/> <input type="button" value="&gt;"/> <input type="button" value="&lt;&lt;"/> <input type="button" value="&gt;&gt;"/>
Internal Users		curiouswave_ad	
Guest Users			
All_AD_Join_Points			

---

**Advanced Search List Settings**

If a selected identity store cannot be accessed for authentication

Do not access other stores in the sequence and set the "AuthenticationStatus" attribute to "ProcessError"

Treat as if the user was not found and proceed to the next store in the sequence

## Integrate AD and Import Groups

Identity Services Engine Home Context Visibility Operations Policy Administration Work

System Identity Management Network Resources Device Portal Management pxGrid Services Feed Service Th

Identities Groups External Identity Sources Identity Source Sequences Settings

**External Identity Sources**

- Certificate Authentication Profile
- Active Directory
  - curiouswave\_ad
- LDAP
- ODBC
- RADIUS Token
  - DUO
  - RSA SecurID

Connection	Whitelisted Domains	PassiveID	Gr
<b>WARNING:</b> This machine must be joined on the connections tab for this feature to t			
/ Edit + Add X Delete Group Update SID Values			
<input type="checkbox"/>	Name		SID
<input type="checkbox"/>	curiouswaves.com/Users/Domain Computers		S-1-
<input type="checkbox"/>	curiouswaves.com/Users/Domain Users		S-1-
<input type="checkbox"/>	curiouswaves.com/Users/IT		S-1-
<input type="checkbox"/>	curiouswaves.com/Users/SALES		S-1-

Built normal TACACS authentication and authorization policy pointing to the source sequence created above. You can get creative here; all I want is to give a user belonging to IT group privilege level 15 if the user passes MFA.

Policy Sets → Default Reset Policyset Hitcounts | Reset | Save

Status	Policy Set Name	Description	Conditions	Allowed Protocols / Server Sequences	Hits
✔	Default	Tacacs Default policy set		Default Device Admin	15

Authentication Policy (1)

Status	Rule Name	Conditions	Use	Hits	Actions
✔	Default		DKO_AD_AUTH	28	Options

Authorization Policy (2)

Status	Rule Name	Conditions	Results	Command Sets	Shell Profiles	Hits	Actions
✔	WLC_AUTH	curiouswave_sd ExternalGroups EQUALS curiouswaves.com/users/IT	PERMIT_ALL		PRV15	6	
✔	Default		DenyAllCommands		Deny All Shell Profile	2	

### 3. Configure Network Infrastructure Device

I'm using basic TACACS config for AAA; you can get as creative as you want. I'm using a very basic TACACS configuration below:

```
pod1_9800CL#sh run | sec aaa|tacacs
aaa new-model
aaa group server tacacs+ TACACS_SRV_GROUP
server name SERVER1
!
tacacs server SERVER1
address ipv4 192.168.129.10
key *****
!
aaa authentication login TACACS-ISE group TACACS_SRV_GROUP local
aaa authorization exec TACACS-ISE group TACACS_SRV_GROUP local
aaa authorization config-commands
aaa authorization commands 1 TACACS-ISE group TACACS_SRV_GROUP local
aaa authorization commands 15 TACACS-ISE group TACACS_SRV_GROUP local
```

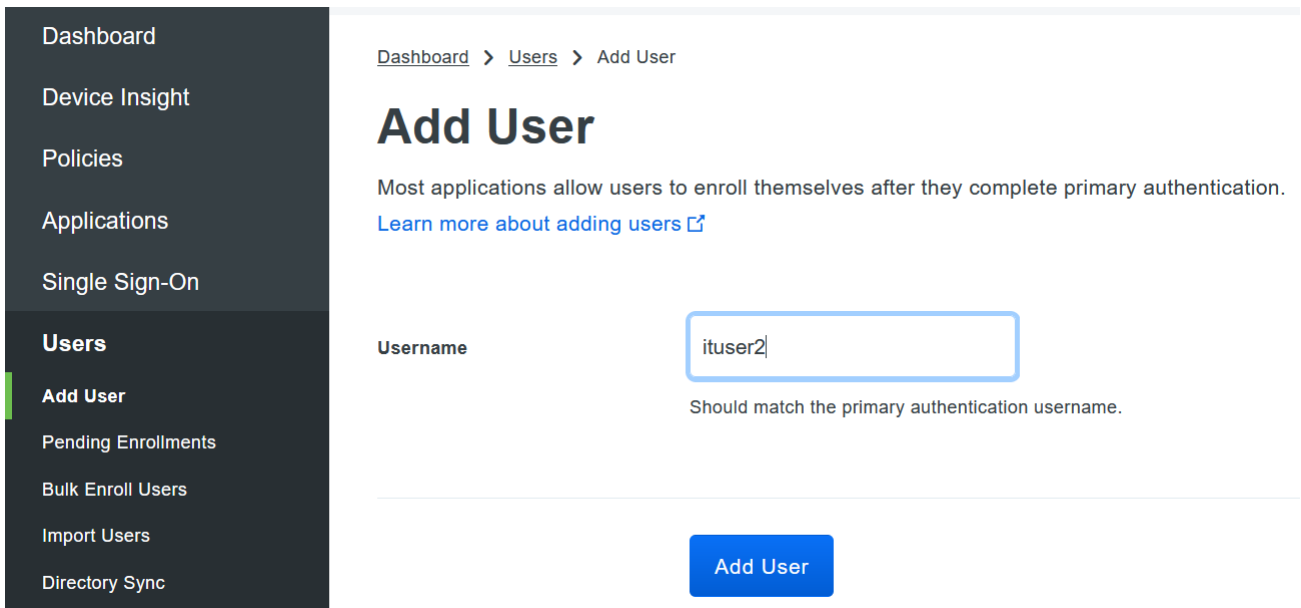
## 4. Provision End User to Use Duo for TACACS Login

In my case, I'm configuring a user manually on Duo. However, if you want to sync AD group with Duo, you can [follow instructions here](#).

It's a three-step process to activate a user:

1. Add the user account with phone number
2. Send activation instructions on phone
3. Install Duo app and follow intrusions to onboard device

### Start Adding User Account and Phone Number




The screenshot shows the Duo Admin console interface. On the left is a dark sidebar with navigation options: Dashboard, Device Insight, Policies, Applications, Single Sign-On, **Users**, Add User, Pending Enrollments, Bulk Enroll Users, Import Users, and Directory Sync. The main content area has a breadcrumb trail: Dashboard > Users > Add User. The title is 'Add User'. Below the title is a paragraph: 'Most applications allow users to enroll themselves after they complete primary authentication.' followed by a link: 'Learn more about adding users'. There is a form field for 'Username' containing the text 'ituser2'. Below the field is a note: 'Should match the primary authentication username.' At the bottom right of the form area is a blue button labeled 'Add User'.



[Dashboard](#) > [Users](#) > [ituser2](#) > Add Phone

# Add Phone

 [Learn more about Activating Duo Mobile](#).

Type  Phone  Tablet


Phone number  [Show extension field](#)  
Optional. Example: "+1 201-555-5555"

[Add Phone](#)

## Send Activation Instructions to Phone

[Dashboard](#) > [Phones](#) > Phone: [redacted]

[Send SMS Passcodes...](#) |  [Delete Phone](#)

 ituser2  
[redacted]

[Attach a user](#)  
Authentication devices  
can share multiple  
users

### Device Info

[Learn more about Activating Duo Mobile](#).

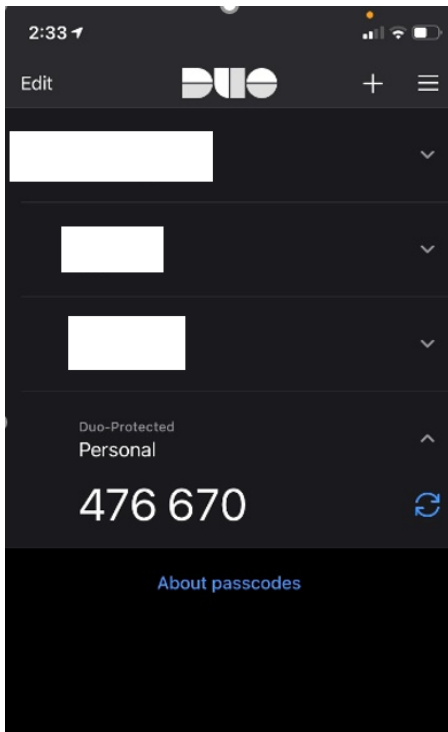
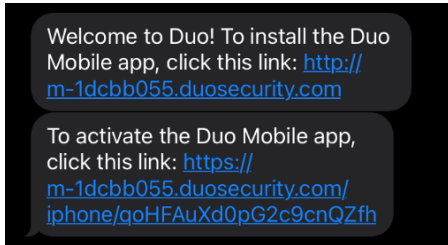
 Not using Duo Mobile  
[Activate Duo Mobile](#)

 **Model**  
Unknown

 **OS**  
Generic Smartphone

## User Receives Instructions and Installs the Duo App

There are multiple ways to authenticate a user; we are using push notification.



## Verify on the Duo Admin Portal that User is Onboarded

Dashboard > Phones > Phone: [REDACTED]



Send SMS Passcodes...



ituser2

[Attach a user](#)

Authentication devices  
can share multiple  
users

### Device Info

[Learn more about Activating Duo Mobile](#)



Using Duo Mobile 3.60.0.10  
[Reactivate Duo Mobile](#)

Last seen  
29 seconds ago



Model  
Apple iPhone [REDACTED]



OS  
iOS 14.7.1

### Device Security



Tampered  
No

[What is a tampered device?](#)



Passcode set  
Yes



Biometrics  
Touch ID or Face ID enabled

We are ready to access our device using MFA! As we do, we can verify the logs on ISE, Auth Proxy and Duo.

On ISE under TACACS live logs we see authentication and authorization logs.

Cisco Identity Services Engine							
Home		Context Visibility	Operations	Policy	Administration	Work Centers	
RADIUS	Threat-Centric NAC Live Logs	TACACS	Troubleshoot	Adaptive Network Control	Reports		
Live Logs							
Refresh Export To							
Logged Time	Status	Details	Identity	Type	Authentication Policy	Authorization Policy	
Sep 11, 2021 02:42:51.371 PM	✓	🔒	ituser2	Authorization	Authentication Policy	Authorization Policy	
Sep 11, 2021 02:42:51.257 PM	✓	🔒	ituser2	Authentication	Default >> Default	Default >> WLC_AUTH	

### AuthProxy Logs Returning Access-Accept

```
[duoauthproxy.lib.log#info] (('192.168.129.10', 32676), ituser2, 26): Duo authentication returned 'allow': 'Success. Logging you
[duoauthproxy.lib.log#info] (('192.168.129.10', 32676), ituser2, 26): Returning response code 2: AccessAccept
[duoauthproxy.lib.log#info] (('192.168.129.10', 32676), ituser2, 26): Sending response
```

## On Duo ituser2 Granted Access

### Authentication Log Last 10 attempts

[Full authentication log](#)

Timestamp (UTC)	Result	User	Application	Access Device	Second Factor
6:42:51 PM SEP 11, 2021	✔ <b>Granted</b> User approved	ituser2	RADIUS	Location Unknown 0.0.0.0	➤ Duo Push Fort Mill, SC, United States

## Conclusion

You saw in this article how to start protecting your critical infrastructure in a few simple steps. However, this is just the tip of the iceberg!

If you have any questions or would like help setting up MFA, please reach out to your DSI account manager or email [sales@dsitech.com](mailto:sales@dsitech.com). They can put you in touch with me directly and we can discuss how to protect your applications/users and network infrastructure with Cisco Duo.

Thank you for reading and we look forward to discussing a new topic in the next newsletter!

## Resources

[Duo Capabilities](#)

[Duo Proxy Supported Devices & Install](#)

[Jailbroken iPhone](#)

[Largest Collection of Password Breaches](#)

[Sync AD Groups with Duo](#)

## About the Author

Ambuj M. is a Cisco Certified Internetwork Expert (CCIE) and Certified Wireless Network Expert (CWNE) with 15 years of industry experience. He currently works as a Network Solutions Architect for DISYS Solutions Inc. (DSI).