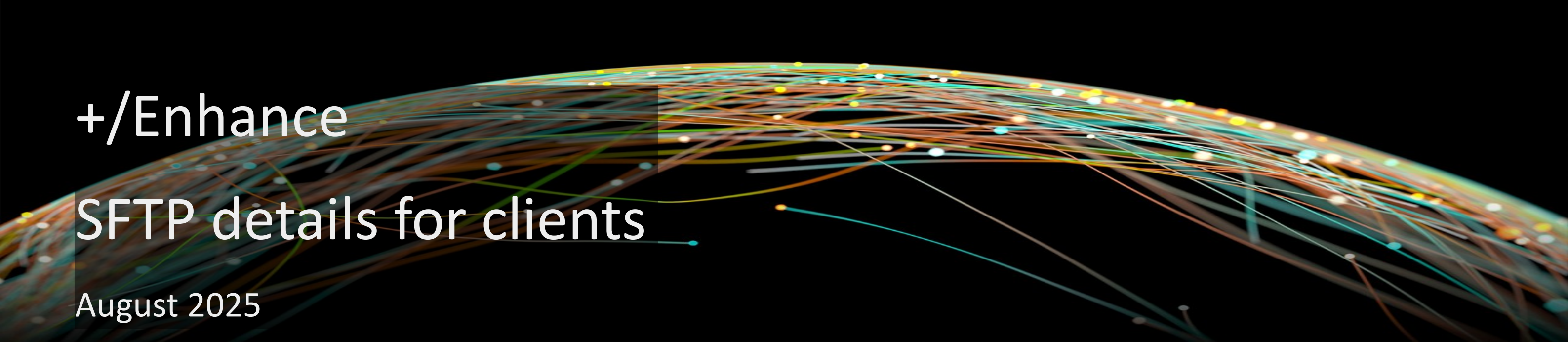

+/Enhance

SFTP details for clients

August 2025

Enhance Group use a service hosted by Amazon's AWS Transfer Family, detailed at https://aws.amazon.com/aws-transfer-family/
This is a secure file transfer facility based on SFTP — a function of the SSH2 protocol suite standard.
This SFTP should not be confused with any similarly named variations of the much older & insecure FTP standard.

Access requires SSH client software[1] such as listed at https://en.wikipedia.org/wiki/Comparison_of_SSH_clients
You are responsible for your client software selection, installation, maintenance & support. Many of those listed at the link above are free & well documented with online tutorials etc. but you can also contact us if you need more help connecting to our service.

You may not need all of the generic details below, depending on your circumstances. Your specific account credentials are overleaf.
Please use copy & paste to get these details into your systems. Do not rely on screen-shot images or verbal dictation.
If you have any questions or require any additional information please use the appropriate contact listed.

Things you will need to know:

Protocol Version	SSH v2 only (default since v1 was downgraded)
Protocol Command	SFTP only (SCP & SSH commands are not supported)
Our Remote Host/End-Point FQDN[2]	`s-6a24d6e2c42c49c98.server.transfer.eu-west-3.amazonaws.com`
Our SFTP service IP address[3]	See note. Currently: `15.236.149.214,15.236.160.151,52.47.172.188`
Our Remote Host Port	22 (default)
Our Host Key Fingerprint (MD5)	`ssh-rsa 2048 f6:d7:c8:82:3e:60:c3:6a:7d:ed:7b:7f:c9:6d:6c:cc`
Our Host Key Fingerprint (SHA256)	`ssh-rsa 2048 sjupj9ctaxozTUixPfTzsYT1NIXXHOq5ieVsIDlh37Y=`
Your Username (overleaf)	(case sensitive, all lower-case, no spaces, allocated by us)
Your Password (overleaf)	(case sensitive, machine generated, of good strength & automation-friendly)

Things you may need to know:

Cipher & Mac	`aes256-ctr` ..&.. `hmac-sha2-256` (required by GoAnywhere.com)
Your Remote Home Folder[4]	/ (Please **put** your files into "**destination**" & **get** our files from "**source**" sub-folders)
Your Filesystem Permissions[5]	List/Read/Write/Delete files/folders. You cannot set timestamp or permissions
Your Public SSH User Key location[6]	/.ssh/authorized_keys (optional OpenSSH format, case sensitive, all lower-case)
Technical contact e-mail	`"Phill Rogers"<PRogers@Enhance.Group>` (real person)
Production contact e-mail	`DataAutomation@Enhance.Group` (distribution group)
Our Public PGP key (rsa2048)	On request. Fingerprint: `E1D7F3F22001D5141C3E94E0E780285C91FA967B`
Purge policy	Each party should purge any files they control after successful collection.
Our Time Zone & countries of relevance[7]	GMT/BST (UTC+01:00) & Jersey (ISO3166:JE), United Kingdom (ISO3166:GB)

Things you need to tell us, with examples:

The public IP address you connect from	10.30.50.70
Technical contact e-mail	`"First Last"<F.Last@YourOrganisation.co.uk>` (E.g. a real person)
Production contact e-mail	`"Operations"<Production@YourOrganisation.co.uk>` (E.g. distribution group)
Your Public SSH User Key	Please upload it or send by e-mail attachment. OpenSSH format.
Your Public PGP key (rsa2048) fingerprint	`DA39A3EE5E6B4B0D3255BFEF95601890AFD80709` (upload or e-mail the actual key)
Your Time Zone & countries of relevance[7]	GMT/BST (UTC+01:00) & Jersey (ISO3166:JE), United Kingdom (ISO3166:GB)

Further related documentation is available upon request:
Mosaic - Data Provision Guidelines   Template+Formats+Validation+Notes+Examples
Data_Transfer (SFTP/FTP disambiguation...)   Data_Transfer_testing   DP-notes (TAB vs CSV...)

STRICTLY CONFIDENTIAL

Firstname Lastname
Firstname.Lastname@Company.net
Company.net

Your ref.: Our ref.:

To help identify this specific data feed in any query correspondence, each party should quote the other’s reference above.

Do not expose within sight of a webcam, CCTV, window or other people.
As at 2025-08-11, your specific username & password are, respectively:

`usernme101` ..and.. `Pa55W0rd`

(for comparison: `0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz`)
Please keep these safe.

Notes:

1.) We recommend WinSCP, PuTTY & OpenSSH which each have varying degrees of user-friendliness & scripting ability.

2.) Please use this rather than IP address. We recommend the use of a secure DNS service (or preferably two) such as DoH (DNS over HTTPS) as listed at https://en.wikipedia.org/wiki/Public_recursive_name_server

3.) Up to date IP addresses are listed at https://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html
Our service does not connect to your end & your connection to our service should validate our host key to ensure you have not reached any other host. Thus there is no real need for you to add the IP of our service to your 'allow-list'. But if you want to then you should create a programmatic method to do this dynamically at the time of connection. As a final option, you could tell your IP allow-list that our IP address (range) is "0.0.0.0/24" which should work fine.
We can notify you of any IP changes if you subscribe to an automatic e-mail.

4.) Folder capacity may be subject to storage quota limits. Please follow good housekeeping.
If your organization has multiple departments sharing this account then each should use their own named sub-folders as appropriate so that E.g. "/put/One Department/" and "/put/Another Department/" can co-exist without conflict. If you have multiple legal entities under the same global domain name then we recommend using separate SFTP accounts for each to maintain jurisdictional separation. Such entities will need to agree distinguishing "Our ref." & "Your ref." names for reference in all related communications.

5.) If using WinSCP, you should either configure it so that it does not try to change permissions &/ timestamp after uploading, or to ignore errors. https://winscp.net/eng/docs/ui_transfer_custom#upload

6.) If you want to use an SSH user key pair, please tell us when you have uploaded your public key, so we can activate it.

7.) The list of countries allows us to find all public holidays which might impact expected delivery.

Our public PGP key for encrypting files to us and verifying the signature of files from us, is as follows:

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
mQENBF7OHSUBCAC5Ygm4KbOMzVsLIMgUXbd8dMbwHs4zqp0QsiA/vwrRRC7q8aal
KVF2tbKD3D+POUF2VikBGrRG5lpYNH0BBq1DluBOQefb9d5TEyCXO+1xR8e8jigg
WHn/vYaibP+22T5NP00ROisO49QOZSJlg5EbmEbWYM+MzkXLuWnwP7acUEKiVB53
9rWKFmPTTauqyKf8CHa8lPCKIkeOJEFuUKLSPtGwEg1S9F2ZqJxFooHiLWwNeMIw
xHvm1x4U8gHjWYcH101jTDKBbmg6gpSW/eDZaeioVyjDh1XDgWSWhKp18/cbphjm
D8w+cOx2HEgp8CGqDL9sy46lQz2ednCC5AOdABEBAAG0SU1vc2FpYyAyIFN5c3Rl
bSBBZG1pbmlzdHJhdG9yIChNMiBHZW5lcmFsKSA8c2FkbWluaXN0cmF0b3JAZW5o
YW5jZS5ncm91cD6JAU4EEwEIADgWIQTh1/PyIAHVFBw+lODngChckfqWewUCXs4d
JQIbAwULCQgHAgYVCgkICwIEFgIDAQIeAQIXgAAKCRDngChckfqWewPnCACoqo2U
d3PdFprqAXhJg/nEHlr+DeJFyERTE4ODPAlnpx6unMkdeI3D3jLDZopp+kxSi0G6
rlBe28NWHuZrxpmfJ36bKjhZR6i4O5eCP6Y4KI81zYv2zXfQhDFCbmDBOgWI0e/R
nFB/o1MQGB8rTK5/e02BI/SW7+lKoi2l45eu24Co/t28kGm8eoWFA4DI2Bpi9Klu
2zbNbAGtL/2n+w0AC7YZAFDcHLMQT26WpXuQdU5I+XHrQNlLkQ9AiROFG3p6M007
p1HjjyTzgLrbi786un7Gb+looG8iozIevJF9tWMsIBcpRx562p1FeWyaLTTSSH41
OXlDtNjGPk+sjv/luQENBF7OHSUBCAC8s4gvaLoXQFvZgvi5R4b5zK7tvI/QAqWB
4JnowairP52+vbcU/SrWr+PC+T19GgFZCpsG/fKtFjsZgOrdbHYpyeHFk2eQukYy
n90gP1rMsUVv+NzzMnSLFBtUXQyhCO6PVlxn/T/ciSFCnjNPNjeyzcOVdo+SUgNO
FTZ1j+AJT3Pbs60CmwgM8jxmz7KoIuuJQTtJ5BbrHFyz2+ru64ykvs4SwuOQWB9X
0f7K5ech9AmQgPjtXiI+xqmQHsENdNp3KhboO+DiwiL0j6yjMEGbnAdkpLTOHJlA
oKcmy9UU4KGSJadPpCLOqf9SJiQRZPzGCKTTeqb0j09+N+nVVvOZABEBAAGJATYE
GAEIACAWIQTh1/PyIAHVFBw+lODngChckfqWewUCXs4dJQIbDAAKCRDngChckfqW
e0byB/0TnYEcmNpp7wd7b1ivNLGjBS4SMoZFOSIxQthWwvWADA+Mja+kS4CL2wG7
YgVewJPdgSMtrQTQ2KUoWjPyOMdWkCg9LGq+NtRRgCY5QxjG8D05nTwH62DyMJ4e
mF0moAr8/GPU4GSWjSxPqu9cx0Eu29pacUTbB2jP4hOVpVzwJPXHocnxzpQ9Adkn
yhN7Xg8RKJ9xzB9EeGv2f37DyTc55YLXGagO3ExrAi/AV4uAPpJVlzkdW9hNQXo/
2+dNhP26zxFOKAKKfLmEg6NrxfcXjKDL32+P21U8MwzdhBkCB4bKcVc22K458HUw
eRfo3kXHMKxRa0YxTAONgjQHEbC4
=GAX6
-----END PGP PUBLIC KEY BLOCK-----
```

SFTP details for service hosts

For when we initiate a connection with an SFTP service which you host.

Things we need to know from you, with examples:

The business unit / jurisdiction / entity	GlobalOrg-Jersey-Custody (when you globally use a common domain name)
Your Remote Host/End-Point FQDN	mft.YourOrganisation.co.us
The public IP address of your service	10.30.50.70
Your public SSH Host Key Fingerprint	(in MD5 or SHA256 format as per our example on the first page)
Our Username	enhancegroup
Our Password	(send by a mutually convenient, secure means such as https://OneTimeSecret.com/)
Technical contact e-mail	"First Last"<F.Last@YourOrganisation.co.uk> (E.g. a real person)
Production contact e-mail	"Operations"<Production@YourOrganisation.co.uk> (E.g. distribution group)
Our Remote Home Folder	/ (or /destination and /source for example)
Our Filesystem Permissions	Can we List/Read/Write/Delete files/folders? Do you auto-purge upon collection?
Your Public PGP key (rsa2048) fingerprint	DA39A3EE5E6B4B0D3255BFEF95601890AFD80709 (and send the actual key)
Your Time Zone & countries of relevance[7]	GMT/BST (UTC+01:00) & Jersey (ISO3166:JE), United Kingdom (ISO3166:GB)

Things we need to tell you:

Our IP address for your allow-list	See note at https://mft.enhance.je/~/MFT_IP.pgp.txt Currently: 51.107.13.232
Technical contact e-mail	"Phill Rogers"<PRogers@Enhance.Group> (real person)
Production contact e-mail	DataAutomation@Enhance.Group (distribution group)
Our Public PGP key (rsa2048) fingerprint	E1D7F3F22001D5141C3E94E0E780285C91FA967B (see actual key above)
Our 2k public SSH user key fingerprint	2048 SHA256:ffX5kBmzqiDaOnAzpUVPkaNIBWBi6/T9XMqSOUVS4/4
Our 2k public SSH user key	See below left. (Save the long string in authorized_keys file, preceded with "ssh-rsa ")
Our 4k public SSH user key fingerprint	4096 SHA256:rwV/UqmP+Y+Lwx/w5XcHruS+5Eyt2VyZ6waZNAvhUe4
Our 4k public SSH user key	See below right. (Save the long string in authorized_keys file, preceded with "ssh-rsa ")

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "rsa-key-20201126 for SFTP access from EnhanceGroup user"
AAAAB3NzaC1yc2EAAAABJQAAAQEAzOf2EXHP6bAmo7Gh9e+0QJJUHr6yJkOoQG40
71pT5LgyZzJT7KrMH5cW9dW4lXqgok5psXk0BFXZvcsSifRKslW0g5AgAYX+no4l
80CRXgtVFYaUPe9FP2sJ5abiXhk3f9z7UsoyL1UevOU0un9wQNLc4VQ2GAzY2fWZ
YKVHyrB6cWOZaFIDubNEo964DGCEq2ZPbkX7A96cxNJdU6zE+Y9kzAv7MI4tWxDW
/9JMmuMrvSnXIzCoHVi0OiRMt+hZk06Ef8BXoezELzuaPHSEfjwprUf99W7E9gn6
H0y/QkmVUTxsW+aoqxZSk05kIrcbOavCTbS3suVumgA2W9Ksfw==
---- END SSH2 PUBLIC KEY ----
```

```
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, by eg\admin-progers@egl-p-avd-0 from"
AAAAB3NzaC1yc2EAAAADAQABAAACAQCrqcCH4UjyCN0ImBisU7mdJZdjF1dutKcqmiisvz
jO7xEaNn9AKOzTFGaWLe7ZIDbC2DVKT4PdVPcO8iqGA543euk40oiEKfmWqVOLnTLAOWyv
W0x8xEb/1RU9/NUeF7gTKPplUe7/WiV/q+weBfOOzy6nzHND7lT9X/GD8g3iSe1gFPi8oS
bYFaQN5DOelWH9XolZlEOMIjYTCFqALBIL4C2DR80VVCVGJCpi7VRM5ylzL4SytB/O8yrE
nHN/8R1SxbSLZPW7n2aa539hdS0ShLRl0XWX9eqSQeXcyQpFKucyGQxjszN6+R8foOyUag
Vf5kgDeCMdt0WAi9iqgF4uMxsX5qL58OXxgblUvr485aHLfccZd9OPr5cwkgHcNbyCbxI5
edijuqFfoaCUx3+wHvJVIRY/bmZ8uD1ITbVBqAWg8b1hii3ecB60lCYgsO+chEWJtYbYS5
EQkXptzwUIOqusWmv3pHMa8bH0sNS2XbzYyPELRSP7bA51oub3YH+7oQI3h8Z7nvyHFbSc
n2h3OL04xQl63CU14eg+EcsdUgFjt3T31t45y/8hJa6vG/Ee5SZ4rYFidq5bXBE2GxthsZ
eGvBz91FxI8O2Q1YKNRw1cbA8W2mZC3UvfZnemGGWtaZ68G2XHTdhb1ELr9RlmPWDBSsGn
B/kDfvME5kjrIQ==
---- END SSH2 PUBLIC KEY ----
```

Why to use fingerprints

The SFTP/SSH protocol provides two-way authentication between client & server.
When implemented correctly, this means that **both** of the following are true:

The host service can be certain that the user connecting is genuine.
The user can be certain the service they are connecting with is genuine.

The host service may be configured to allow the client to authenticate their username with a few different methods, including: a password, a key pair, a hardware device, or some combination.

The user's client software is always able to authenticate the server by the host key pair.
Even if user authentication is only by password, the key-based server authentication is always there.
And it doesn't matter if the user is pushing files for upload or pulling files for download with the service.

Important note: The SSH **host** key pair and the SSH **user** key pair are two different things.
The user key can prove to the host service that the user connecting is genuine.
The host key can prove to the user that the service they are connecting with is genuine.

This is fundamentally different from standard FTP for example, where the connecting user has no way to tell if they have reached the genuine service or an imposter.

Without the second part of this two-way authentication, the communication between parties is at risk of attack by a "*man in the middle*" imposter, which is then able to intercept and copy or replace any data being transferred. Without even having to break any encryption!

The Internet was fundamentally designed to allow easy re-routing of traffic to cope with equipment or connection failures. This attribute can easily be abused by a malicious actor to spoof an IP address or a domain name, enabling them to receive connection requests intended for others.

If the user initiating the connection doesn't realise, they will then send their sensitive login credentials to the imposter. The imposter then uses those credentials and initiates its own connection with the genuine intended service provider. Perhaps spoofing the genuine user's IP address. From this point on, the imposter simply forwards any data from one end to the other and both genuine parties think they have a secure connection. But any data between them is actually at the mercy of the imposter.

A secure DNS service such as DoH (DNS over HTTPS) and filtering IP addresses by an "allow-list" both help but neither can prevent this completely. This is exactly what the host key authentication is for, and why it is important to use it properly.

During onboarding, the operators of the host service should provide a conveniently short fingerprint of their public host key, which the user can compare with the fingerprint detected by their client software on first connection. If the fingerprints match, the connecting user can tell their client software to only accept a connection with this matching fingerprint from now on.

When a user first connects to a new service, their SFTP client software will show them the fingerprint(s) of the server they have reached. Whether it is the genuine one or an imposter. If the user blindly accepts this, then they will always be able to connect with this server. Whether it is the genuine one or an imposter.

If the fingerprint for an imposter was accepted, then any subsequent connection to the genuine server will fail, probably causing a re-try. So even if the imposter is only occasionally able to intercept the connection, it becomes certain that the client only successfully connects with the imposter.

The public part of the SSH host key pair can be represented by a short and convenient "fingerprint" in any of a few different formats.
It will look something like:

```
    MD5 format:       ssh-rsa 2048 f6:d7:c8:82:3e:60:c3:6a:7d:ed:7b:7f:c9:6d:6c:cc
    SHA256 format:    ssh-rsa 2048 sjupj9ctaxozTUixPfTzsYT1NIXXHOq5ieVsIDlh37Y=
```

If you are hosting an SFTP/SSH service and have not offered or been asked for your host key fingerprint then any files sent or received by your client users could have been copied or replaced without either party being aware. Take a break for a chat with your compliance officer if you need to.