SSH

Tutorial

Γιώργος Καππές

SSH Tutorial

Το ssh (Secure Shell) είναι ένα ασφαλές δικτυακό πρωτόκολλο το οποίο επιτρέπει τη μεταφορά δεδομένων μεταξύ δύο υπολογιστών. Το ssh όχι μόνο κρυπτογραφεί τα δεδομένα που ανταλλάσσονται κατά τη συνεδρία, αλλά προσφέρει ένα ασφαλές σύστημα αναγνώρισης καθώς και άλλα χαρακτηριστικά όπως ασφαλές μεταφορά αρχείων κλπ.

Ξεκινώντας

Το πρώτο πράγμα που θα δούμε είναι πως μπορούμε απλά να συνδεθούμε σε ένα απομακρυσμένο σύστημα το οποίο ανήκει είτε στο υποδίκτυό μας, είτε στο internet. Έστω, λοιπόν, ότι θέλουμε να συνδεθούμε στο Scylla.cs.uoi.gr θα πρέπει να εκτελέσουμε:

```
# ssh scylla.cs.uoi.gr
```

Το ssh θα χρησιμοποιήσει το όνομα χρήστη του λογαριασμού από τον οποίο το εκτελούμε για να συνδεθεί στον απομακρυσμένο υπολογιστή. Αν θέλετε να συνδεθείτε με ένα διαφορετικό όνομα μπορείτε να εκτελέσετε το παρακάτω:

```
# ssh username @scylla.cs.uoi.gr
```

Και στις δύο περιπτώσεις ο απομακρυσμένος υπολογιστής θα μας ζητήσει τον κωδικό πρόσβασης που μας έχει ανατεθεί στο απομακρυσμένο σύστημα. Την πρώτη φορά που επιχειρούμε να συνδεθούμε μέσω του ssh σε έναν απομακρυσμένο υπολογιστή θα μας ζητηθεί εάν θέλουμε να εισάγουμε τον απομακρυσμένο υπολογιστή στη λίστα με τους γνωστούς υπολογιστές. Είναι σημαντικό να δώσουμε προσοχή σε αυτή την ερώτηση διότι πρόκειται για ένα από τα σημαντικότερα χαρακτηριστικά του ssh το host validation. Το χαρακτηριστικό αυτό σας επιτρέπει να είστε σίγουροι ότι συνδέεστε στον επιθυμητό υπολογιστή κι όχι σε κάποιον άλλο που απλά παριστάνει τον συγκεκριμένο υπολογιστή:

```
The authenticity of host Scylla.cs.uoi.gr (193.92.4.7) can't be established.
RSA key fingerprint is 53:b4:ad:c8:51:17:99:4b:c9:08:ac:c1:b6:05:71:9b.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'arvo.suso.org' (RSA) to the list of known
```

Εάν κάποιος προσπαθήσει να σας ωθήσει στο να συνδεθείτε στο σύστημά του προσποιούμενος ότι το σύστημά του αντιστοιχεί σε αυτό που θέλετε να συνδεθείτε, το ssh θα σας ενημερώσει με το ακόλουθο μήνυμα:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@       WARNING: POSSIBLE DNS SPOOFING DETECTED!          @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

The RSA host key for arvo.suso.org has changed,
and the key for the according IP address 216.9.137.122
is unchanged. This could either mean that
DNS SPOOFING is happening or the IP address for the host
and its host key have changed at the same time.
Offending key for IP in /home/suso/.ssh/known_hosts:10
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@

IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
96:92:62:15:90:ec:40:12:47:08:00:b8:f8:4b:df:5b.
Please contact your system administrator.
Add correct host key in /home/suso/.ssh/known_hosts to get rid of this message.
Offending key in /home/suso/.ssh/known_hosts:53
RSA host key for arvo.suso.org has changed and you have requested strict
checking.
Host key verification failed.
```

Εάν δείτε αυτό το μήνυμα θα πρέπει να σταματήσετε και να σκεφτείτε εάν υπάρχει κάποιος λόγος για την αλλαγή του host key στο απομακρυσμένο σύστημα. Το προτιμότερο είναι να επικοινωνήσετε με τον διαχειριστή του συστήματος πριν συνεχίσετε τη συνεδρία.

Δημιουργήστε ένα keyfile

Τώρα που έχετε δει κάποια από τα βασικά χαρακτηριστικά του ssh θα σας δείξουμε πώς μπορείτε να δημιουργήσετε ένα keyfile έτσι ώστε να συνδέεστε με μεγαλύτερη ασφάλεια στο απομακρυσμένο σύστημα. Αποσυνδεθείτε από το απομακρυσμένο σύστημα και δώστε στο τερματικό:

```
# ssh-keygen -t dsa
```

Θα σας ζητηθεί η θέση όπου θα αποθηκευτεί το κλειδί. Αν δεν έχετε ήδη δημιουργήσει κάποιο κλειδί, πατήστε enter για να επιλέξετε την προεπιλεγμένη θέση. Στη συνέχεια θα σας ζητηθεί το passphrase. Το passphrase είναι κάτι σαν τον κωδικό πρόσβασης, με τη

διαφορά ότι μπορείτε να χρησιμοποιήσετε μια ολόκληρη πρόταση. Μη χρησιμοποιείτε συνηθισμένες προτάσεις όπως ρητά.

Καθώς δημιουργείτε ένα κλειδί, στην πραγματικότητα δημιουργούνται δύο κλειδιά, ένα public και ένα private. Το private κλειδί πρέπει να μείνει στο σύστημά σας και πρέπει να προσέχετε να μη πέσει σε άλλους. Το public κλειδί, που προφανώς είναι διαφορετικό από το private πρέπει να το τοποθετήσετε στο απομακρυσμένο σύστημα. Το public κλειδί μπορεί να το δει οποιοσδήποτε και είναι μαθηματικά αδύνατο να χρησιμοποιηθεί από κάποιον για την παραγωγή του private κλειδιού. Για να μεταφέρετε το public κλειδί στο απομακρυσμένο σύστημα κάντε τα εξής:

1. Συνδεθείτε στο απομακρυσμένο σύστημα με ssh
2. Δημιουργήστε στον φάκελο .ssh στον home κατάλογό σας εάν αυτό δεν υπάρχει
3. Δώστε τα κατάλληλα δικαιώματα πρόσβασης:

```
# chmod 700 ~/.ssh
# chmod 600 ~/.ssh/authorized_keys
```

Αποσυνδεθείτε από το απομακρυσμένο σύστημα και εκτελέστε:

```
# scp ~/.ssh/id_dsa.pub username@arvo.suso.org:.ssh/authorized_keys
```

Πλέον το public κλειδί σας βρίσκεται στον απομακρυσμένο υπολογιστή. Καθώς θα συνδέεστε στον απομακρυσμένο υπολογιστή από εδώ και στο εξής, θα εξετάζεται το private και το public key και θα σας ζητείται η εισαγωγή του passphrase (Προσοχή! Όχι του κωδικού!). Η μέθοδος αυτή σας προφυλάσσει από επιθέσεις spoofing. Για παράδειγμα κάποιος cracker μπορεί να κλέψει το αρχείο κωδικών του απομακρυσμένου server όπου θέλετε να συνδεθείτε και παραπλανώντας το dns, να σας ξεγελάσει έτσι ώστε να συνδεθείτε στο δικό του σύστημα. Η επίθεση αυτή αποτυγχάνει στην περίπτωση που χρησιμοποιείτε δημόσιο/ιδιωτικό κλειδί, αφού για να πραγματοποιηθεί η συνεδρία δε λαμβάνεται υπόψη ο κωδικός πρόσβασης στο απομακρυσμένο σύστημα αλλά τα κλειδιά.

X11 Session forwarding

Ένα από τα κύρια χαρακτηριστικά των X windows που χρησιμοποιούνται κατά κόρον στα Unix συστήματα είναι η δικτυακή διαφάνεια που παρέχουν. Σχεδιάστηκα έτσι ώστε να μεταφέρουν τις πληροφορίες των παραθύρων και των ιδιοτήτων τους μέσα στο δίκτυο δίνοντας τη δυνατότητα σε απομακρυσμένους χρήστες να έχουν πρόσβαση στην επιφάνεια εργασίας του συστήματος. Έτσι μπορείτε να συνδεθείτε στην επιφάνεια εργασίας ενός απομακρυσμένου συστήματος και να τρέξετε παραθυρικές εφαρμογές όπως το Gimp, το gedit κλπ. Μπορείτε να χρησιμοποιήσετε το ssh για να συνδεθείτε σε μια απομακρυσμένη επιφάνεια εργασίας με ασφάλεια ως εξής:

```
# ssh -X username@desktopmachine.domain.com
```

Με την παραπάνω εντολή προωθείτε τη σύνδεση X μέσω της σύνδεσης ssh. Για να πετύχει βέβαια η σύνδεση πρέπει στον απομακρυσμένο υπολογιστή να τρέχει ο κατάλληλος δαίμονας. Για να το πετύχετε αυτό τροποποιείστε το αρχείο /etc/ssh/sshd_config στον απομακρυσμένο υπολογιστή (πρέπει αν έχετε δικαιώματα root) ως εξής:

```
X11Forwarding yes
X11DisplayOffset 10
X11UseLocalhost yes
```

Για κάποια νέα προγράμματα και κάποιες νέες εκδόσεις των X Windows μπορεί να χρειαστεί να χρησιμοποιήσετε την παρακάτω εντολή για να συνδεθείτε:

```
ssh -Y username@desktopmachine.domain.com
```

Χρησιμοποιείστε αυτή την επιλογή εάν η συνεδρία σας αρνηθεί την αίτηση σύνδεσης με την επιλογή –X και σας επιστραφεί ένα μήνυμα που μοιάζει με το παρακάτω:

```
The program ‘gimp-2.2’ received an X Window System error.
This probably reflects a bug in the program.
The error was ‘BadWindow (invalid Window parameter)’.
(Details: serial 154 error_code 3 request_code 38 minor_code 0)
(Note to programmers: normally, X errors are reported asynchronously;
that is, you will receive the error a while after causing it.
To debug your program, run it with the—sync command line
option to change this behavior. You can then get a meaningful
backtrace from your debugger if you break on the gdk_x_error()
function.)
```

TCP Port Forwarding

Το ssh μπορεί επίσης να προωθήσει συνδέσεις TCP στο επίπεδο εφαρμογών από και προς τη συνεδρία ssh. Το TCP port forwarding χωρίζεται στις εξής κατηγορίες:

Local Port forwarding

Χρησιμοποιείται στις εξής περιπτώσεις:

- Όταν θέλουμε να προωθήσουμε μη ασφαλές συνδέσεις TCP μέσω του ssh για μεγαλύτερη ασφάλεια. Για παράδειγμα, μπορούμε να ασφαλίσουμε συνδέσεις POP3, SMTP, HTTP, κλπ.

- Όταν θέλουμε να παρακάμψουμε κάποιο firewall στο υποδίκτυο που βρισκόμαστε, το οποίο δε μας επιτρέπει να χρησιμοποιήσουμε μια ορισμένη υπηρεσία παρά μόνο το ssh.

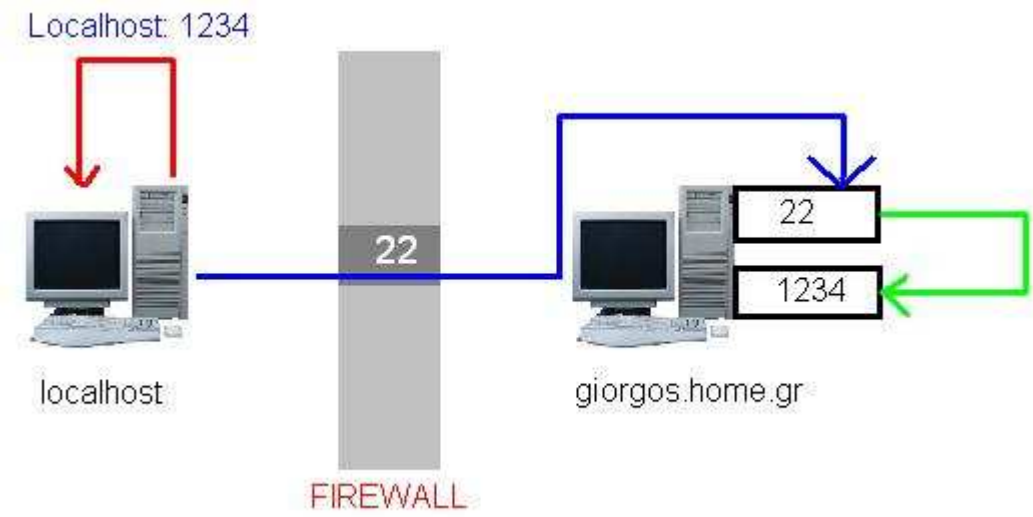
Για να χρησιμοποιήσουμε το Port forwarding θα χρειαστεί να ανοίξουμε ορισμένες θύρες στο σύστημά μας. Για να ανοίξουμε θύρες μικρότερα από την θύρα 1024 θα πρέπει να έχουμε root access στα αντίστοιχα μηχανήματα, έτσι εμείς θα χρησιμοποιήσουμε ports μεαλύτερα του 1024 για το σύστημα localhost.

Σενάριο 1

Έστω ότι βρισκόμαστε στον υπολογιστή localhost και θέλουμε να συνδεθούμε στη θύρα 1234 του απομακρυσμένου μηχανήματος giorgos.home.gr όπου εκτελείτε κάποια υπηρεσία. Όμως το firewall πίσω από το οποίο βρίσκεται ο υπολογιστής localhost δε μας επιτρέπει να χρησιμοποιήσουμε τη θύρα 1234 παρά μόνο τη θύρα 22 (ssh). Στην περίπτωση αυτή μπορούμε να εκτελέσουμε:

```
# ssh -L 1234:localhost:1234 user@giorgos.home.gr
```

όπου στην προκειμένη περίπτωση, ανοίγει μια port στο τοπικό μηχάνημα και στο παράδειγμά μας η port 1234 και ότι σύνδεση γίνει εκεί μεταφέρεται αμέσως μέσω του ssh και κάνει σύνδεση στο giorgos.home.gr στην port 1234.



(Σχήμα 1)

Πλέον, αρκεί να ρυθμίσουμε τον επιθυμητό μας client έτσι ώστε να χρησιμοποιεί σαν server το local μηχάνημα και την port 1234. Αυτή η μέθοδος μπορεί επίσης να μας βοηθήσει στο να ασφαλίσουμε ορισμένες μη ασφαλές TCP συνδέσεις.

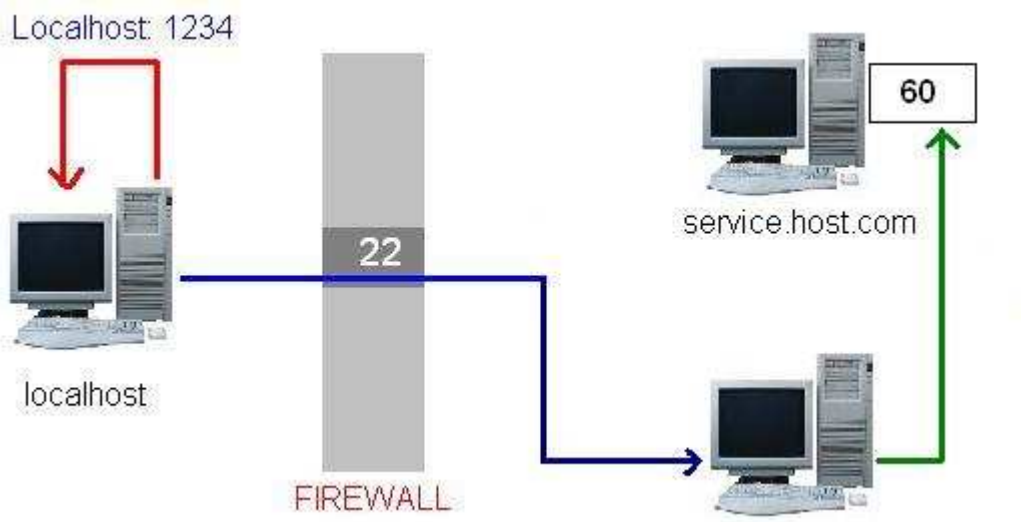
Σενάριο 2

Έστω ότι βρισκόμαστε στο μηχάνημα localhost και θέλουμε να συνδεθούμε σε μια υπηρεσία του μηχανήματος service.host.com στην πόρτα 60. Όμως το firewall πίσω από το

οποίο βρισκόμαστε δε μας επιτρέπει τη συγκεκριμένη σύνδεση, παρά μόνο το ssh. Ας υποθέσουμε λοιπόν ότι έχουμε ανοικτό ένα τρίτο μηχάνημα, το home.myhome.com στο οποίο μπορούμε να συνδεθούμε με ssh. Θα συνδεθούμε, λοιπόν, στη θύρα 60 του service.host.com χρησιμοποιώντας σαν ενδιάμεσο σταθμό το μηχάνημα home. myhome.com:

```
# ssh –L 1234:service.host.com:60 user@home.myhome.gr
```

Τι έχουμε πετύχει; Έχουμε συνδεθεί στο μηχάνημα service.host.com παρακάπτοντας το firewall. Πλέον, αρκεί να ρυθμίσουμε τον επιθυμητό μας client έτσι ώστε να χρησιμοποιεί σαν server το local μηχάνημα και την port 1234. Σημειώστε ότι το μηχάνημα service.host.com δε βλέπει το τοπικό μας μηχάνημα, αλλά το home.myhome.gr.



(σχήμα 2)

Για παράδειγμα μπορούμε να συνδεθούμε σε έναν IRC server ως εξής:

```
# ssh -L 1080:uk.irc.gr:6667 my.linux.server.gr
```

όπου στην προκειμένη περίπτωση, ανοίγει μια port στο τοπικό μηχάνημα και στο παράδειγμά μας η port 1080 και ότι σύνδεση γίνει εκεί μεταφέρεται αμέσως μέσω του ssh και κάνει σύνδεση στο server uk.irc.gr στην port 6667. Με το συγκεκριμένο παράδειγμα μπορούμε να ανοίξουμε τον αγαπημένο μας IRC Client και να δώσουμε σαν server το local μηχάνημα και port την 1080:

```
/server 127.0.0.1 1080
```

και να συνδεθούμε στο GRNet και συγκεκριμένα στον server uk.irc.gr. Ομοίως, μπορούμε να κάνουμε το ίδιο και για ένα web site ή ένα ftp server.

Remote Port Forwarding

Αυτό θα ήτανε ιδιαίτερα χρήσιμο αν θέλουμε να συνδεθεί κάποιος προσωρινά στο μηχάνημα μας ή σε κάποιο μηχάνημα στο τοπικό μας δίκτυο. Με την εντολή:

```
# ssh -R 5100:192.168.0.3:80 my.linux.server.gr
```

ανοίγουμε την port 5100 στο μηχάνημα my.linux.server.gr και του ορίζουμε πως θα ανακατευθύνει ότι συνδέσεις γίνονται εκεί μέσω του ssh στο μηχάνημα 192.168.0.3 και στην port 80 (που είναι οι web υπηρεσίες συνήθως). Έν ολίγοις αν στο μηχάνημα μας 192.168.0.3 έχουμε κάποιο web server, όποιος χρήστης επισκεφθεί τον ιστότοπο http://my.linux.server.gr:5100/ στην πραγματικότητα θα δει τον web server μας. Η παραπάνω μέθοδος μπορεί να χρησιμοποιηθεί ώστε να ρυθμίζουμε το router του σπιτιού μας από ένα απομακρυσμένο μηχάνημα. Έστω ότι το router δέχεται συνδέσεις στην τοπική διεύθυνση 192.168.2.1 μέσω της θύρας 80. Εκτελώντας:

```
# ssh -R 5100:192.168.2.1:80 my.linux.server.gr
```

κάνουμε τη σελίδα ρυθμίσεων του router ορατή μέσω του internet στη διεύθυνση: http://my.linux.server.gr:5100/

SOCKS Proxying

Με τη βοήθεια του ssh μπορείτε να χρησιμοποιήσετε ένα απομακρυσμένο μηχάνημα σαν proxy για κάθε εφαρμογή που υποστηρίζει το SOCKS (firefox, gaim, xchat κ.α.). Για να το πετύχετε αυτό καθορίζεται μια θύρα στο τοπικό σας μηχάνημα όπου θα προωθούνται οι συνδέσεις:

```
# ssh -D 9999 username@remotehost.net
```

και ρυθμίζεται το πρόγραμμα πελάτη σας να χρησιμοποιεί ως proxy το localhost στην πόρτα 9999. Πλέον οι απομακρυσμένοι υπολογιστές στους οποίους θα συνδέεστε θα βλέπουν το μηχάνημα remotehost.net και όχι εσάς!

Εκτελέστε μια εντολή μέσω του ssh

Εάν θέλετε να εκτελέσετε μια εντολή σε ένα απομακρυσμένο μηχάνημα εκτελέστε:

```
# ssh username@remotehost.net εντολή
```