

Exercici 1

Creem un fitxer a l'arrel del nostre disc dur, i executem el codi:

```
public class Exercici2u1a1 {  
    public static void main(String[] args) throws Exception {  
        File fichero = new File("C:\\Test.txt");  
        FileReader fr = new FileReader(fichero);  
        BufferedReader buffer = new BufferedReader(fr);  
        String cad;  
        while ((cad = buffer.readLine()) != null) {  
            System.out.println(cad);  
        }  
    }  
}
```

ut - Exercici2u1 (run)

```
]Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "C:\\Test.txt" "read")  
    at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)  
    at java.security.AccessController.checkPermission(AccessController.java:884)  
    at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)  
    at java.lang.SecurityManager.checkRead(SecurityManager.java:888)  
    at java.io.FileInputStream.<init>(FileInputStream.java:127)  
    at java.io.FileReader.<init>(FileReader.java:72)  
    at exercici2u1a1.Exercici2u1a1.main(Exercici2u1a1.java:15)  
C:\\Users\\Eric\\AppData\\Local\\NetBeans\\Cache\\8.2\\executor-snippets\\run.xml:53: Java returned: 1
```

Com podem veure no tenim permisos per accedir al fitxer, per lo que creem un java.policy a l'arrel del projecte i li fem la següent línia per obtenir permisos:

permission java.security.AllPermission;

permission java.io.FilePermission "C:/Test.txt", "read";

Podem seleccionar si volem donar-li tots els permisos o només donar-li permisos al fitxer en qüestió i seleccionar quin tipus, en el meu cas li he donat de lectura únicament.

Si ens fixem ara si que deixa:

```
run:  
fefwreferwf  
BUILD SUCCESSFUL (total time: 0 seconds)
```

Exercici 2

Per aquest exercici volem comprovar si les diferents IP's estan accessibles per nosaltres.

Per això provem si funciona el codi del programa:

```
ut - Exercici2u1 (run)
```

```
run:  
]Exception in thread "main" java.security.AccessControlException: access denied ("java.util.PropertyPermission" "user.dir" "read")  
    at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)  
    at java.security.AccessController.checkPermission(AccessController.java:884)  
    at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)  
    at java.lang.SecurityManager.checkPropertyAccess(SecurityManager.java:1294)  
    at java.lang.System.getProperty(System.java:717)  
    at exercici2u1a2.Exercici2u1a2.main(Exercici2u1a2.java:26)  
C:\\Users\\Eric\\AppData\\Local\\NetBeans\\Cache\\8.2\\executor-snippets\\run.xml:53: Java returned: 1  
BUILD FAILED (total time: 0 seconds)
```

Com podem veure no tenim permisos per accedir a les polítiques de la nostra carpeta, per lo que donem permisos per poder llegir dins de la nostra:

permission java.util.PropertyPermission "user.dir", "read";

Tornem a provar i ara ens diu que no tenim permisos pel FileSeparator per lo que el fiquem a un String com a SystemProperty i li donem permisos de propietats:

```
it - Exercici2u1 (run)
run:
Exception in thread "main" java.security.AccessControlException: access denied ("java.util.PropertyPermission" "file.separator" "read")
    at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)
    at java.security.AccessController.checkPermission(AccessController.java:884)
    at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)
    at java.lang.SecurityManager.checkPropertyAccess(SecurityManager.java:1294)
    at java.lang.System.getProperty(System.java:717)
    at exercici2ula2.Exercici2ula2.main(Exercici2ula2.java:27)
C:\Users\Eric\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 0 seconds)
```

permission java.util.PropertyPermission "file.separator", "read";

Ara ens falten permisos per accedir al fitxer, per lo que li donem permisos de fitxer:

```
ut - Exercici2u1 (run)
run:
] Exception in thread "main" java.security.AccessControlException: access denied ("java.io.FilePermission" "C:\Users\Eric\Desktop\DAW\M9\UF1\Exercici2ula1\names.txt" "read")
    at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)
    at java.security.AccessController.checkPermission(AccessController.java:884)
    at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)
    at java.lang.SecurityManager.checkRead(SecurityManager.java:888)
    at java.io.File.isFile(File.java:877)
    at exercici2ula2.Exercici2ula2.main(Exercici2ula2.java:30)
C:\Users\Eric\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 0 seconds)
```

permission java.io.FilePermission "names.txt", "read, write";

Per últim un cop solucionat tot lo anterior amb les diferents sentencies, veiem que tenim una restricció amb els permisos del socket:

```
[- Exercici2u1 (run)
run:
Exception in thread "main" java.security.AccessControlException: access denied ("java.net.SocketPermission" "192.168.120.20:80" "connect, resolve")
    at java.security.AccessControlContext.checkPermission(AccessControlContext.java:472)
    at java.security.AccessController.checkPermission(AccessController.java:884)
    at java.lang.SecurityManager.checkPermission(SecurityManager.java:549)
    at java.lang.SecurityManager.checkConnect(SecurityManager.java:1051)
    at java.net.Socket.connect(Socket.java:584)
    at java.net.Socket.connect(Socket.java:538)
    at java.net.Socket.<init>(Socket.java:434)
    at java.net.Socket.<init>(Socket.java:211)
    at exercici2ula2.Exercici2ula2.main(Exercici2ula2.java:38)
C:\Users\Eric\AppData\Local\NetBeans\Cache\8.2\executor-snippets\run.xml:53: Java returned: 1
BUILD FAILED (total time: 0 seconds)
```

Per lo que afegim els permisos pels Sockets:







permission java.net.SocketPermission "192.168.120.20:80", "connect, resolve";

I com podem veure ara si que ens busca si les IP si estan accessibles o no:

```
it - Exercici2u1 (run)
run:
/192.168.120.20...AVAILABLE
BUILD SUCCESSFUL (total time: 0 seconds)
```

Exercici 3

Per crear un .jar certificat, es a dir, firmat digitalment, necessitem tenir en una mateixa carpeta, el .jar, el java.policy, el contingut que utilitza l'aplicació (en aquest cas el names.txt) i els certificats, etc. :


 desti.crt	18/05/2017 15:57	Certificado de seg...	1 KB
 desti.jks	18/05/2017 15:59	Archivo JKS	4 KB
 Exercici2u1.jar	25/05/2017 15:16	Executable Jar File	5 KB
 java.policy	25/05/2017 15:33	Archivo POLICY	1 KB
 names.txt	24/05/2017 18:47	Documento de tex...	1 KB
 origen.jks	25/05/2017 15:21	Archivo JKS	4 KB

El .jar generat firmat es genera a partir del certificat de destí i la creació d'un nou.

Després creem un .jar nou però signat a partir del certificat:

```
C:\Users\Eric\Desktop\DAM\M9\UF1\Exercici2u1a1\jar>-keystore desti.jks -signedjar Exercici2u1-signed.jar Exercici2u1.jar  
desti_
```

Com podem veure ens crea el .jar signat :

 Exercici2u1-signer.jar

I si executem podem veure que el .jar signat s'executa correctament:

```
C:\Users\Eric\Desktop\DAM\M9\UF1\Exercici2u1a1\jar>java -Djava.security.manager -Djava.security.policy==java.policy -jar  
Exercici2u1-signer.jar  
/192.168.120.20...AVAILABLE
```

Si intentem executar ara l'antic .jar no ens deixaria, ja que ara tenim el grant limitat per "desti", per lo que si no esta firmat per aquest no es pot executar. Per tal d'executar-lo hauríem de crear un específic o un general per l'altre/s.

```
Keystore "origen.jks";  
grant SignedBy "desti"{
```