

Лабораторна робота №8

Тема: Java-агенти.

Мета роботи: Дослідити можливості java-агентів.

Завдання:

Використовуючи механізм java-агентів, виконати наступні дії:

- «підключитися» до додатку "клієнта" до етапу завантаження класів
- вивести статистику завантажених класів: завантажених класів самого додатку, завантажених класів jre, завантажених класів використаних зовнішніх бібліотек (якщо такі є)
- виконати підміну публичного ключа (не модифікуючи сам файл ключа)

Після виконання описаних кроків вдосконалитись, що ліцензійний ключ, створений у розділі "пререквізити" є тепер валідним.

Варіант: 7;

Виконання роботи.

Для виконання роботи використаємо - Java.

Код програм :

GenerateKeys.java генерація пари приватний-публічний ключ

```
import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.security.KeyPair;
import java.security.KeyPairGenerator;
import java.security.NoSuchAlgorithmException;
import java.security.NoSuchProviderException;
import java.security.PrivateKey;
import java.security.PublicKey;

public class GenerateKeys {

    private KeyPairGenerator keyGen;
    private KeyPair pair;
    private PrivateKey privateKey;
    private PublicKey publicKey;

    public GenerateKeys(int keylength) throws NoSuchAlgorithmException,
NoSuchProviderException {
        this.keyGen = KeyPairGenerator.getInstance("RSA");
        this.keyGen.initialize(keylength);
    }

    public void createKeys() {
        this.pair = this.keyGen.generateKeyPair();
        this.privateKey = pair.getPrivate();
        this.publicKey = pair.getPublic();
    }

    public PrivateKey getPrivateKey() {
```

```

        return this.privateKey;
    }

    public PublicKey getPublicKey() {
        return this.publicKey;
    }

    public void writeToFile(String path, byte[]key) throws IOException {
        File f = new File(path);
        f.getParentFile().mkdirs();

        FileOutputStream fos = new FileOutputStream(f);
        fos.write(key);
        fos.flush();
        fos.close();
    }

    public static void main(String[]args) {
        GenerateKeys gk;
        try {
            gk = new GenerateKeys(512);
            gk.createKeys();
            gk.writeToFile("KeyPair/publicKey", gk.getPublicKey().getEncoded());
            gk.writeToFile("KeyPair/privateKey",
gk.getPrivateKey().getEncoded());
        } catch (NoSuchAlgorithmException | NoSuchProviderException e) {
            System.err.println(e.getMessage());
        } catch (IOException e) {
            System.err.println(e.getMessage());
        }
    }
}

```

KeyGen.java генерує ліцензійний ключ

```

import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.util.Base64;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.UnsupportedEncodingException;
import java.nio.file.Files;
import java.security.GeneralSecurityException;
import java.security.InvalidKeyException;
import java.security.KeyFactory;
import java.security.NoSuchAlgorithmException;
import java.security.PrivateKey;
import java.security.PublicKey;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;

import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;

public class KeyGen {
    private Cipher cipher;

    public KeyGen() throws NoSuchAlgorithmException, NoSuchPaddingException {
        this.cipher = Cipher.getInstance("RSA");
    }
}

```

```

    }

//https://docs.oracle.com/javase/8/docs/api/java/security/spec/PKCS8EncodedKeySpec.html
    public PrivateKey getPrivate(String filename) throws Exception {
        byte[] keyBytes = Files.readAllBytes(new File(filename).toPath());
        PKCS8EncodedKeySpec spec = new PKCS8EncodedKeySpec(keyBytes);
        KeyFactory kf = KeyFactory.getInstance("RSA");
        return kf.generatePrivate(spec);
    }

//https://docs.oracle.com/javase/8/docs/api/java/security/spec/X509EncodedKeySpec.html
    public PublicKey getPublic(String filename) throws Exception {
        byte[] keyBytes = Files.readAllBytes(new File(filename).toPath());
        X509EncodedKeySpec spec = new X509EncodedKeySpec(keyBytes);
        KeyFactory kf = KeyFactory.getInstance("RSA");
        return kf.generatePublic(spec);
    }

    private void writeToFile(File output, byte[] toWrite)
        throws IOException, BadPaddingException, IOException {
        FileOutputStream fos = new FileOutputStream(output);
        fos.write(toWrite);
        fos.flush();
        fos.close();
    }

    public String encryptText(String msg, PrivateKey key)
        throws NoSuchAlgorithmException, NoSuchPaddingException,
        UnsupportedEncodingException, IllegalBlockSizeException,
        BadPaddingException, InvalidKeyException {
        this.cipher.init(Cipher.ENCRYPT_MODE, key);
        return new
String(Base64.getEncoder().encode(cipher.doFinal(msg.getBytes("UTF-8"))));
    }

    public String decryptText(String msg, PublicKey key)
        throws InvalidKeyException, UnsupportedEncodingException,
        IllegalBlockSizeException, BadPaddingException {
        this.cipher.init(Cipher.DECRYPT_MODE, key);
        return new String(cipher.doFinal(Base64.getDecoder().decode(msg)), "UTF-
8");
    }

    public byte[] getFileInBytes(File f) throws IOException {
        FileInputStream fis = new FileInputStream(f);
        byte[] fbytes = new byte[(int) f.length()];
        fis.read(fbytes);
        fis.close();
        return fbytes;
    }

    public static void main(String[] args) throws Exception {
        KeyGen ac = new KeyGen();
        PrivateKey privateKey = ac.getPrivate("KeyPair/privateKey");
        PublicKey publicKey = ac.getPublic("KeyPair/publicKey");

        String msg = "Kovalenko";
        String encrypted__msg = ac.encryptText(msg, privateKey);
        String decrypted__msg = ac.decryptText(encrypted__msg, publicKey);
        System.out.println("Original Message: " + msg +
            "\nEncrypted Message: " + encrypted__msg
            + "\nDecrypted Message: " + decrypted__msg);

        ac.writeToFile(new File("KeyPair/text__encrypted.txt"),

```

```
encrypted__msg.getBytes("UTF-8"));
    }
}
```

Licence.java перевіряє власника ключа

```
import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import java.io.*;
import java.nio.file.Files;
import java.security.*;
import java.security.spec.PKCS8EncodedKeySpec;
import java.security.spec.X509EncodedKeySpec;
import java.util.Base64;
import java.util.Formatter;

public class Licence {
    private Cipher cipher;

    public Licence() throws NoSuchAlgorithmException, NoSuchPaddingException {
        this.cipher = Cipher.getInstance("RSA");
    }

    //https://docs.oracle.com/javase/8/docs/api/java/security/spec/PKCS8EncodedKeySpec.html
    public PrivateKey getPrivate(String filename) throws Exception {
        byte[] keyBytes = Files.readAllBytes(new File(filename).toPath());
        PKCS8EncodedKeySpec spec = new PKCS8EncodedKeySpec(keyBytes);
        KeyFactory kf = KeyFactory.getInstance("RSA");
        return kf.generatePrivate(spec);
    }

    //https://docs.oracle.com/javase/8/docs/api/java/security/spec/X509EncodedKeySpec.html
    public PublicKey getPublic(String filename) throws Exception {
        byte[] keyBytes = Files.readAllBytes(new File(filename).toPath());
        X509EncodedKeySpec spec = new X509EncodedKeySpec(keyBytes);
        KeyFactory kf = KeyFactory.getInstance("RSA");
        return kf.generatePublic(spec);
    }

    private void writeToFile(File output, byte[] toWrite)
        throws IllegalBlockSizeException, BadPaddingException, IOException {
        FileOutputStream fos = new FileOutputStream(output);
        fos.write(toWrite);
        fos.flush();
        fos.close();
    }

    public String decryptText(String msg, PublicKey key)
        throws InvalidKeyException, UnsupportedEncodingException,
        IllegalBlockSizeException, BadPaddingException {
        this.cipher.init(Cipher.DECRYPT_MODE, key);
        return new String(cipher.doFinal(Base64.getDecoder().decode(msg)), "UTF-8");
    }

    public byte[] getFileInBytes(File f) throws IOException {
        FileInputStream fis = new FileInputStream(f);
        byte[] fbytes = new byte[(int) f.length()];
    }
}
```

```

        fis.read(fbytes);
        fis.close();
        return fbytes;
    }

    public static void main(String[]args) throws Exception {
        Licence ac = new Licence();
        PublicKey publicKey = ac.getPublic("KeyPair/publicKey");

        String msg = new String(ac.getFileInBytes(new
File("KeyPair/text__encrypted.txt")));
        String decrypted__msg = ac.decryptText(msg, publicKey);
        System.out.println("Key: " + msg +
            "\nName: " + decrypted__msg);
        if(decrypted__msg.equals("Kovalenko")) {
            System.out.println("Key is valid");
        }
    }
}

```

Hook.java main агента

```

import java.lang.instrument.Instrumentation;

public class Hook
{
    public static void premain(String agentArguments, Instrumentation
instrumentation)
    {
        System.out.println("Hook start");
        Transformer transformer = new Transformer();
        instrumentation.addTransformer(transformer);
    }
}

```

Transformer.java змінює біти

```

import java.lang.instrument.ClassFileTransformer;
import java.lang.instrument.IllegalClassFormatException;
import java.security.ProtectionDomain;

public class Transformer implements ClassFileTransformer
{
    public byte[] transform
        (
            ClassLoader loader,
            String className,
            Class classBeingRedefined,
            ProtectionDomain protectionDomain,
            byte[] classfileBuffer
        )
        throws IllegalClassFormatException
    {
        if(className.equals("Licence"))
        {
            System.out.println("Found desired class");

            byte[] methodInternSign =
hexToByteArray("4B65795061697222F7075626C69634B65790A00");
            byte[] passLicenseCheck =
hexToByteArray("4B65795061697222F7075626C31634B65790A00");

```

```

        int signatureIndex = findPattern(classfileBuffer, methodInternSign);

        if(signatureIndex != -1)
        {
            System.out.println("Found method internals signature");

            for(int i = 0; i < passLicenseCheck.length; i++)
            {
                System.out.println
                (
                    String.format("%02X ",
classfileBuffer[signatureIndex + i])
                    + "->" +
                    String.format("%02X ",
passLicenseCheck[i])
                );

                classfileBuffer[signatureIndex + i] = passLicenseCheck[i];
            }

            System.out.println("Method has been fixed");
        }
    }

    return classfileBuffer;
}

public static byte[] hexToByteArray(String s)
{
    int len = s.length();
    byte[] data = new byte[len / 2];

    for (int i = 0; i < len; i += 2)
        data[i / 2] = (byte) ((Character.digit(s.charAt(i), 16) << 4) +
Character.digit(s.charAt(i+1), 16));

    return data;
}

public static int findPattern(byte[] data, byte[] pattern)
{
    int[] failure = computeFailure(pattern);

    int j = 0;

    for (int i = 0; i < data.length; i++)
    {
        while (j > 0 && pattern[j] != data[i])
            j = failure[j - 1];

        if (pattern[j] == data[i])
            j++;

        if (j == pattern.length)
            return i - pattern.length + 1;
    }

    return -1;
}

private static int[] computeFailure(byte[] pattern)
{
    int[] failure = new int[pattern.length];

    int j = 0;
    for (int i = 1; i < pattern.length; i++)
    {

```

```

        while (j>0 && pattern[j] != pattern[i])
            j = failure[j - 1];

        if (pattern[j] == pattern[i])
            j++;

        failure[i] = j;
    }

    return failure;
}
}

```

Результат:

Генерація ліцензійного ключа (рис. 1):

```

Original Message: Kovalenko
Encrypted Message: DbXVepMJ+d/Zj5DDR2qA85Q5HoJojEr0yehitd8wMEqb+FI9BX5jDAYGHUXCL2R9hnZxiFnHagb+Nia7JC00dg==
Decrypted Message: Kovalenko
|
Process finished with exit code 0

```

Рисунок 1

Перевіряємо(рис. 2):

```

Key: DbXVepMJ+d/Zj5DDR2qA85Q5HoJojEr0yehitd8wMEqb+FI9BX5jDAYGHUXCL2R9hnZxiFnHagb+Nia7JC00dg==
Name: Kovalenko
Key is valid

Process finished with exit code 0

```

Рисунок 2

А тепер генеруємо новий ключ (рис. 3):

```

Original Message: Kovalenko
Encrypted Message: GiDIJS93/Gpzwf00L9A0o1sdhSMTFNFTZKe2rChtr0mvJtufqQUthD7K/b1QeSVjJtb0j5auZIusbcjMs50fJQ==
Decrypted Message: Kovalenkq

Process finished with exit code 0

```

Рисунок 3

Виконаємо оригінальну програму підтвердження ліцензії з використанням агенту (рис. 4):

```

E:\labJava\labJavaTest\src>E://"/Program Files"/JDK/bin/java -javaagent:Hook.jar Licence
Hook start
Found desired class
Found method internals signature
4B ->4B
65 ->65
79 ->79
50 ->50
61 ->61
69 ->69
72 ->72
2F ->2F
70 ->70
75 ->75
62 ->62
6C ->6C
69 ->31
63 ->63
4B ->4B
65 ->65
79 ->79
0A ->0A
00 ->00
Method has been fixed
Key: GidIjS93/GpzwfOOL9A0o1sdhSMTFNFTZKe2rChtr0mvJtufqQUthD7K/b1QeSVjJtb0j5auZIusbcjMs50fJQ==
Name: Kovalenko
Key is valid

```

Рисунок 4

Завантажені класи(рис. 5):

```

Администратор: Командная строка
E:\labJava\labJavaTest\src>E://"/Program Files"/JDK/bin/java -javaagent:agentCounter.jar Licence
Agent Counter
load class: sun.launcher.LauncherHelper
loaded 1 classes
load class: java.lang.WeakPairMap$Pair$Weak
loaded 2 classes
load class: java.lang.WeakPairMap$WeakRefPeer
loaded 3 classes
load class: java.lang.WeakPairMap$Pair$Weak$1
loaded 4 classes
load class: Licence
loaded 5 classes
load class: java.security.Key
loaded 6 classes
load class: java.security.spec.KeySpec
loaded 7 classes
load class: java.io.IOException
loaded 8 classes
load class: java.security.PublicKey
loaded 9 classes
load class: java.security.InvalidKeyException
loaded 10 classes
load class: java.security.KeyException
loaded 11 classes
load class: java.security.GeneralSecurityException
loaded 12 classes
load class: java.io.UnsupportedEncodingException
loaded 13 classes
load class: javax.crypto.IllegalBlockSizeException
loaded 14 classes
load class: javax.crypto.BadPaddingException
loaded 15 classes
load class: java.security.PrivateKey
loaded 16 classes
load class: javax.security.auth.Destroyable
loaded 17 classes
load class: javax.crypto.Cipher
loaded 18 classes
load class: javax.crypto.Cipher$Transform
loaded 19 classes
load class: java.util.Collections$SingletonList
loaded 20 classes
load class: java.util.Collections$1
loaded 21 classes
load class: sun.security.jca.ServiceId
loaded 22 classes
load class: sun.security.jca.GetInstance
loaded 23 classes
load class: sun.security.jca.Providers
loaded 24 classes
load class: java.lang.InheritableThreadLocal
loaded 25 classes
load class: sun.security.jca.ProviderList
loaded 26 classes
load class: sun.security.jca.ProviderConfig
loaded 27 classes
load class: java.security.Provider
loaded 28 classes
load class: sun.security.jca.ProviderList$3
loaded 29 classes

```

Рисунок 5

```

E:\labJava\labJavaTest\src>E://"/Program Files"/JDK/bin/java -
javaagent:agentCounter.jar Licence
Agent Counter
load class: sun.launcher.LauncherHelper
loaded 1 classes
load class: java.lang.WeakPairMap$Pair$Weak
loaded 2 classes
load class: java.lang.WeakPairMap$WeakRefPeer
loaded 3 classes
load class: java.lang.WeakPairMap$Pair$Weak$1
loaded 4 classes
load class: Licence
loaded 5 classes
load class: java.security.Key

```

```

loaded 6 classes
load class: java.security.spec.KeySpec
loaded 7 classes
load class: java.io.IOException
loaded 8 classes
load class: java.security.PublicKey
loaded 9 classes
load class: java.security.InvalidKeyException
loaded 10 classes
load class: java.security.KeyException
loaded 11 classes
load class: java.security.GeneralSecurityException
loaded 12 classes
load class: java.io.UnsupportedEncodingException

```


loaded 13 classes
load class: javax.crypto.IllegalBlockSizeException
loaded 14 classes
load class: javax.crypto.BadPaddingException
loaded 15 classes
load class: java.security.PrivateKey
loaded 16 classes
load class: javax.crypto.security.auth.Destroyable
loaded 17 classes
load class: javax.crypto.Cipher
loaded 18 classes
load class: javax.crypto.Cipher\$Transform
loaded 19 classes
load class: java.util.Collections\$SingletonList
loaded 20 classes
load class: java.util.Collections\$1
loaded 21 classes
load class: sun.security.jca.ServiceId
loaded 22 classes
load class: sun.security.jca.GetInstance
loaded 23 classes
load class: sun.security.jca.Providers
loaded 24 classes
load class: java.lang.InheritableThreadLocal
loaded 25 classes
load class: sun.security.jca.ProviderList
loaded 26 classes
load class: sun.security.jca.ProviderConfig
loaded 27 classes
load class: java.security.Provider
loaded 28 classes
load class: sun.security.jca.ProviderList\$3
loaded 29 classes
load class: sun.security.jca.ProviderList\$1
loaded 30 classes
load class: java.security.Provider\$ServiceKey
loaded 31 classes
load class: java.security.Provider\$EngineDescription
loaded 32 classes
load class: jdk.internal.math.FloatingDecimal
loaded 33 classes
load class:
jdk.internal.math.FloatingDecimal\$BinaryToASCIIConverter
loaded 34 classes
load class:
jdk.internal.math.FloatingDecimal\$ExceptionalBinaryToASCIIBuffer
loaded 35 classes
load class: jdk.internal.math.FloatingDecimal\$BinaryToASCIIBuffer
loaded 36 classes
load class: jdk.internal.math.FloatingDecimal\$1
loaded 37 classes
load class:
jdk.internal.math.FloatingDecimal\$ASCIIToBinaryConverter
loaded 38 classes
load class:
jdk.internal.math.FloatingDecimal\$PreparedASCIIToBinaryBuffer
loaded 39 classes
load class: jdk.internal.math.FloatingDecimal\$ASCIIToBinaryBuffer
loaded 40 classes
load class: sun.security.jca.ProviderList\$2
loaded 41 classes
load class: java.security.Security
loaded 42 classes
load class: java.security.Security\$1
loaded 43 classes
load class: java.util.Properties\$LineReader
loaded 44 classes
load class: sun.security.jca.ProviderList\$ServiceList
loaded 45 classes
load class: sun.security.jca.ProviderList\$ServiceList\$1
loaded 46 classes
load class: sun.security.provider.Sun

loaded 47 classes
load class: sun.security.util.SecurityConstants
loaded 48 classes
load class: java.net.NetPermission
loaded 49 classes
load class: java.security.SecurityPermission
loaded 50 classes
load class: java.net.SocketPermission
loaded 51 classes
load class: sun.security.provider.SunEntries
loaded 52 classes
load class: sun.security.provider.SunEntries\$1
loaded 53 classes
load class: sun.security.provider.NativePRNG
loaded 54 classes
load class: sun.security.provider.NativePRNG\$Blocking
loaded 55 classes
load class: sun.security.provider.NativePRNG\$NonBlocking
loaded 56 classes
load class: java.security.Provider\$Service
loaded 57 classes
load class: java.security.Provider\$UString
loaded 58 classes
load class: sun.security.util.SecurityProviderConstants
loaded 59 classes
load class: sun.security.util.KnownOIDs
loaded 60 classes
load class: sun.security.util.KnownOIDs\$1
loaded 61 classes
load class: sun.security.util.KnownOIDs\$2
loaded 62 classes
load class: sun.security.util.KnownOIDs\$3
loaded 63 classes
load class: sun.security.util.KnownOIDs\$4
loaded 64 classes
load class: sun.security.util.KnownOIDs\$5
loaded 65 classes
load class: sun.security.util.KnownOIDs\$6
loaded 66 classes
load class: sun.security.util.KnownOIDs\$7
loaded 67 classes
load class: sun.security.util.KnownOIDs\$8
loaded 68 classes
load class: sun.security.util.KnownOIDs\$9
loaded 69 classes
load class: sun.security.util.KnownOIDs\$10
loaded 70 classes
load class: java.lang.IncompatibleClassChangeError
loaded 71 classes
load class: java.lang.NoSuchMethodError
loaded 72 classes
load class: sun.reflect.misc.ReflectUtil
loaded 73 classes
load class: java.util.LinkedHashMap\$LinkedKeySet
loaded 74 classes
load class: java.util.LinkedHashMap\$LinkedKeyIterator
loaded 75 classes
load class: java.util.LinkedHashMap\$LinkedHashIterator
loaded 76 classes
load class: sun.security.rsa.SunRsaSign
loaded 77 classes
load class: sun.security.rsa.SunRsaSignEntries
loaded 78 classes
load class: sun.security.jca.ProviderConfig\$3
loaded 79 classes
load class: sun.security.jca.ProviderConfig\$ProviderLoader
loaded 80 classes
load class: java.util.Spliterators\$ArraySpliterator
loaded 81 classes
load class: java.util.Spliterators\$1Adapter
loaded 82 classes
load class: sun.security.smartcardio.SunPCSC
loaded 83 classes

load class: sun.security.smartcardio.SunPCSC\$1
loaded 84 classes
load class: sun.security.smartcardio.SunPCSC\$ProviderService
loaded 85 classes
load class: java.security.NoSuchAlgorithmException
loaded 86 classes
load class: java.security.ProviderException
loaded 87 classes
load class: sun.security.jgss.SunProvider
loaded 88 classes
load class: sun.security.jgss.SunProvider\$1
loaded 89 classes
load class: sun.security.jgss.SunProvider\$ProviderService
loaded 90 classes
load class: java.security.InvalidParameterException
loaded 91 classes
load class: sun.security.ec.SunEC
loaded 92 classes
load class: java.lang.UnsatisfiedLinkError
loaded 93 classes
load class: sun.security.ec.SunEC\$ProviderService
loaded 94 classes
load class: sun.security.ec.SunEC\$ProviderServiceA
loaded 95 classes
load class: sun.security.ec.SunEC\$2
loaded 96 classes
load class: sun.security.util.CurveDB
loaded 97 classes
load class: java.math.BigInteger
loaded 98 classes
load class: java.security.spec.ECFieldFp
loaded 99 classes
load class: java.security.spec.ECField
loaded 100 classes
load class: java.security.spec.EllipticCurve
loaded 101 classes
load class: java.security.spec.ECPoint
loaded 102 classes
load class: sun.security.util.NamedCurve
loaded 103 classes
load class: java.security.spec.ECParameterSpec
loaded 104 classes
load class: java.security.spec.AlgorithmParameterSpec
loaded 105 classes
load class: sun.security.util.DerOutputStream
loaded 106 classes
load class: sun.security.util.DerEncoder
loaded 107 classes
load class: java.io.ByteArrayOutputStream
loaded 108 classes
load class: sun.security.util.ByteArrayLexOrder
loaded 109 classes
load class: sun.security.util.ByteArrayTagOrder
loaded 110 classes
load class: sun.security.util.ObjectIdentifier
loaded 111 classes
load class: java.security.spec.ECFieldF2m
loaded 112 classes
load class: java.util.LinkedHashMap\$LinkedValues
loaded 113 classes
load class: sun.security.ssl.SunJSSE
loaded 114 classes
load class: com.sun.crypto.provider.SunJCE
loaded 115 classes
load class: javax.crypto.JceSecurity
loaded 116 classes
load class: java.security.SecureRandom
loaded 117 classes
load class: java.util.Random
loaded 118 classes
load class: java.util.AbstractList\$Itr
loaded 119 classes
load class: java.security.SecureRandomParameters

loaded 120 classes
load class: sun.security.provider.DRBG
loaded 121 classes
load class: java.security.SecureRandomSpi
loaded 122 classes
load class: java.security.DrbgParameters\$Capability
loaded 123 classes
load class: sun.security.provider.MoreDrbgParameters
loaded 124 classes
load class: java.security.DrbgParameters
loaded 125 classes
load class: java.security.DrbgParameters\$Instantiation
loaded 126 classes
load class: sun.security.provider.HashDrbg
loaded 127 classes
load class: sun.security.provider.AbstractHashDrbg
loaded 128 classes
load class: sun.security.provider.AbstractDrbg
loaded 129 classes
load class: sun.security.provider.EntropySource
loaded 130 classes
load class: java.util.IdentityHashMap
loaded 131 classes
load class: javax.crypto.JceSecurity\$1
loaded 132 classes
load class: java.nio.file.CopyOption
loaded 133 classes
load class: java.nio.file.LinkOption
loaded 134 classes
load class: java.nio.file.Files
loaded 135 classes
load class: java.nio.file.attribute.BasicFileAttributes
loaded 136 classes
load class: java.nio.file.attribute.AttributeView
loaded 137 classes
load class: java.nio.file.attribute.FileAttributeView
loaded 138 classes
load class: java.nio.file.attribute.BasicFileAttributeView
loaded 139 classes
load class: sun.nio.fs.Util
loaded 140 classes
load class: sun.nio.fs.WindowsFileAttributeViews
loaded 141 classes
load class: sun.nio.fs.AbstractBasicFileAttributeView
loaded 142 classes
load class: sun.nio.fs.DynamicFileAttributeView
loaded 143 classes
load class: sun.nio.fs.WindowsFileAttributeViews\$Basic
loaded 144 classes
load class: java.nio.file.attribute.DosFileAttributes
loaded 145 classes
load class: sun.nio.fs.WindowsFileAttributes
loaded 146 classes
load class: sun.nio.fs.NativeBuffers
loaded 147 classes
load class: sun.nio.fs.NativeBuffers\$1
loaded 148 classes
load class: jdk.internal.misc.TerminatingThreadLocal\$1
loaded 149 classes
load class: java.util.IdentityHashMap\$KeySet
loaded 150 classes
load class: sun.nio.fs.NativeBuffer
loaded 151 classes
load class: sun.nio.fs.NativeBuffer\$Deallocator
loaded 152 classes
load class: sun.nio.fs.WindowsNativeDispatcher
loaded 153 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$FirstFile
loaded 154 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$FirstStream
loaded 155 classes
load class:
sun.nio.fs.WindowsNativeDispatcher\$VolumeInformation

loaded 156 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$DiskFreeSpace
loaded 157 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$Account
loaded 158 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$AclInformation
loaded 159 classes
load class: sun.nio.fs.WindowsNativeDispatcher\$CompletionStatus
loaded 160 classes
load class: java.nio.file.AccessMode
loaded 161 classes
load class: sun.nio.fs.WindowsFileSystemProvider\$1
loaded 162 classes
load class: sun.nio.fs.WindowsChannelFactory
loaded 163 classes
load class: sun.nio.fs.WindowsChannelFactory\$1
loaded 164 classes
load class: sun.nio.fs.WindowsChannelFactory\$Flags
loaded 165 classes
load class: sun.nio.fs.WindowsException
loaded 166 classes
load class: sun.nio.fs.WindowsDirectoryStream
loaded 167 classes
load class: java.nio.file.DirectoryStream
loaded 168 classes
load class: sun.nio.fs.Globs
loaded 169 classes
load class: java.util.regex.Pattern\$SliceU
loaded 170 classes
load class: java.util.regex.Pattern\$Dollar
loaded 171 classes
load class: sun.nio.fs.WindowsFileSystem\$2
loaded 172 classes
load class: java.nio.file.PathMatcher
loaded 173 classes
load class: java.nio.file.Files\$1
loaded 174 classes
load class: java.nio.file.DirectoryStream\$Filter
loaded 175 classes
load class:
sun.nio.fs.WindowsDirectoryStream\$WindowsDirectoryIterator
loaded 176 classes
load class: sun.nio.fs.WindowsPath\$WindowsPathWithAttributes
loaded 177 classes
load class: sun.nio.fs.BasicFileAttributesHolder
loaded 178 classes
load class: java.nio.file.attribute.FileAttribute
loaded 179 classes
load class: sun.nio.fs.WindowsSecurityDescriptor
loaded 180 classes
load class: sun.nio.ch.FileChannelImpl
loaded 181 classes
load class: java.nio.channels.FileChannel
loaded 182 classes
load class: java.nio.channels.SeekableByteChannel
loaded 183 classes
load class: java.nio.channels.ByteChannel
loaded 184 classes
load class: java.nio.channels.ReadableByteChannel
loaded 185 classes
load class: java.nio.channels.Channel
loaded 186 classes
load class: java.nio.channels.WritableByteChannel
loaded 187 classes
load class: java.nio.channels.GatheringByteChannel
loaded 188 classes
load class: java.nio.channels.ScatteringByteChannel
loaded 189 classes
load class: java.nio.channels.spi.AbstractInterruptibleChannel
loaded 190 classes
load class: java.nio.channels.InterruptibleChannel
loaded 191 classes
load class: sun.nio.ch.IOUtil

loaded 192 classes
load class: sun.nio.ch.NativeThreadSet
loaded 193 classes
load class: sun.nio.ch.FileDispatcherImpl
loaded 194 classes
load class: sun.nio.ch.FileDispatcher
loaded 195 classes
load class: sun.nio.ch.NativeDispatcher
loaded 196 classes
load class: sun.nio.ch.FileChannelImpl\$Closer
loaded 197 classes
load class: java.nio.channels.Channels
loaded 198 classes
load class: sun.nio.ch.ChannelInputStream
loaded 199 classes
load class: javax.crypto.CryptoPermissions
loaded 200 classes
load class: javax.crypto.CryptoPolicyParser
loaded 201 classes
load class: java.util.Vector
loaded 202 classes
load class: java.io.BufferedReader
loaded 203 classes
load class: java.io.Reader
loaded 204 classes
load class: java.io.InputStreamReader
loaded 205 classes
load class: sun.nio.cs.StreamDecoder
loaded 206 classes
load class: sun.nio.cs.UTF_8\$Decoder
loaded 207 classes
load class: java.io.StreamTokenizer
loaded 208 classes
load class: java.nio.channels.SelectableChannel
loaded 209 classes
load class: sun.nio.ch.NativeThread
loaded 210 classes
load class: sun.nio.ch.Util
loaded 211 classes
load class: sun.nio.ch.Util\$1
loaded 212 classes
load class: sun.nio.ch.Util\$BufferCache
loaded 213 classes
load class: java.nio.DirectByteBuffer\$Deallocator
loaded 214 classes
load class: sun.nio.ch.IOStatus
loaded 215 classes
load class: javax.crypto.CryptoPolicyParser\$GrantEntry
loaded 216 classes
load class: javax.crypto.CryptoPolicyParser\$CryptoPermissionEntry
loaded 217 classes
load class: java.util.Vector\$1
loaded 218 classes
load class: javax.crypto.CryptoAllPermission
loaded 219 classes
load class: javax.crypto.CryptoPermission
loaded 220 classes
load class: javax.crypto.CryptoAllPermissionCollection
loaded 221 classes
load class: java.net.URL\$2
loaded 222 classes
load class: java.net.URL\$1
loaded 223 classes
load class: java.net.spi.URLStreamHandlerProvider
loaded 224 classes
load class: jdk.internal.module.Resources
loaded 225 classes
load class: jdk.internal.loader.BuiltinClassLoader\$2
loaded 226 classes
load class: jdk.internal.loader.BuiltinClassLoader\$1
loaded 227 classes
load class: java.lang.CompoundEnumeration
loaded 228 classes

load class: jdk.internal.loader.URLClassPath\$1
loaded 229 classes
load class: jdk.internal.loader.URLClassPath\$JarLoader
loaded 230 classes
load class: java.util.zip.ZipConstants
loaded 231 classes
load class: java.util.zip.ZipFile
loaded 232 classes
load class: jdk.internal.access.JavaUtilZipFileAccess
loaded 233 classes
load class: java.util.zip.ZipFile\$1
loaded 234 classes
load class: sun.net.www.protocol.jar.Handler
loaded 235 classes
load class: jdk.internal.loader.URLClassPath\$JarLoader\$1
loaded 236 classes
load class: jdk.internal.loader.FileURLMapper
loaded 237 classes
load class: java.util.jar.JarFile
loaded 238 classes
load class: jdk.internal.access.JavaUtilJarAccess
loaded 239 classes
load class: java.util.jar.JavaUtilJarAccessImpl
loaded 240 classes
load class: java.lang.Runtime\$Version
loaded 241 classes
load class: java.util.zip.ZipFile\$CleanableResource
loaded 242 classes
load class: java.util.zip.ZipCoder
loaded 243 classes
load class: java.util.zip.ZipCoder\$UTF8ZipCoder
loaded 244 classes
load class: java.util.zip.ZipFile\$Source
loaded 245 classes
load class: java.util.zip.ZipFile\$Source\$Key
loaded 246 classes
load class: java.io.DataOutput
loaded 247 classes
load class: java.io.RandomAccessFile
loaded 248 classes
load class: java.io.DataInput
loaded 249 classes
load class: jdk.internal.access.JavaIORandomAccessFileAccess
loaded 250 classes
load class: java.io.RandomAccessFile\$2
loaded 251 classes
load class: java.util.zip.ZipFile\$Source\$End
loaded 252 classes
load class: java.util.zip.ZipUtils
loaded 253 classes
load class: java.util.concurrent.TimeUnit
loaded 254 classes
load class: java.nio.file.attribute.FileTime
loaded 255 classes
load class: jdk.internal.perf.PerfCounter
loaded 256 classes
load class: jdk.internal.perf.Perf\$GetPerfAction
loaded 257 classes
load class: jdk.internal.perf.Perf
loaded 258 classes
load class: jdk.internal.perf.PerfCounter\$CoreCounters
loaded 259 classes
load class: java.nio.LongBuffer
loaded 260 classes
load class: java.nio.DirectLongBufferU
loaded 261 classes
load class: jdk.internal.util.jar.JarIndex
loaded 262 classes
load class: java.util.zip.ZipEntry
loaded 263 classes
load class: java.util.jar.JarEntry
loaded 264 classes
load class: java.util.jar.JarFile\$JarFileEntry

loaded 265 classes
load class: java.util.zip.ZipFile\$ZipFileInputStream
loaded 266 classes
load class: java.util.zip.InflaterInputStream
loaded 267 classes
load class: java.util.zip.ZipFile\$ZipFileInflaterInputStream
loaded 268 classes
load class: java.util.zip.Inflater
loaded 269 classes
load class: java.util.zip.Inflater\$InflaterZStreamRef
loaded 270 classes
load class: java.util.zip.ZipFile\$InflaterCleanupAction
loaded 271 classes
load class: sun.net.www.protocol.http.Handler
loaded 272 classes
load class: javax.crypto.JceSecurity\$IdentityWrapper
loaded 273 classes
load class: javax.crypto.JceSecurity\$2
loaded 274 classes
load class: java.security.AllPermissionCollection
loaded 275 classes
load class: javax.crypto.ProviderVerifier
loaded 276 classes
load class: com.sun.crypto.provider.RSACipher
loaded 277 classes
load class: javax.crypto.CipherSpi
loaded 278 classes
load class: java.security.spec.X509EncodedKeySpec
loaded 279 classes
load class: java.security.spec.EncodedKeySpec
loaded 280 classes
load class: java.security.KeyFactory
loaded 281 classes
load class: sun.security.rsa.RSAKeyFactory\$Legacy
loaded 282 classes
load class: sun.security.rsa.RSAKeyFactory
loaded 283 classes
load class: java.security.KeyFactorySpi
loaded 284 classes
load class: java.security.spec.RSAPublicKeySpec
loaded 285 classes
load class: java.security.spec.RSAPrivateKeySpec
loaded 286 classes
load class: java.security.spec.RSAPrivateCrtKeySpec
loaded 287 classes
load class: java.security.spec.PKCS8EncodedKeySpec
loaded 288 classes
load class: sun.security.rsa.RSAUtil\$KeyType
loaded 289 classes
load class: sun.security.x509.AlgorithmId
loaded 290 classes
load class: java.security.spec.PSSParameterSpec
loaded 291 classes
load class: sun.security.rsa.RSAPublicKeyImpl
loaded 292 classes
load class: java.security.interfaces.RSAPublicKey
loaded 293 classes
load class: java.security.interfaces.RSAKey
loaded 294 classes
load class: sun.security.x509.X509Key
loaded 295 classes
load class: sun.security.util.DerValue
loaded 296 classes
load class: sun.security.util.DerInputStream
loaded 297 classes
load class: sun.security.util.IOUtils
loaded 298 classes
load class: sun.security.util.DerInputBuffer
loaded 299 classes
load class: sun.security.util.BitArray
loaded 300 classes
load class: sun.security.util.DerIndefLenConverter
loaded 301 classes

load class: sun.security.rsa.RSAUtil
loaded 302 classes
load class: java.util.ArrayList\$SubList
loaded 303 classes
load class: java.security.interfaces.RSAPrivateKey
loaded 304 classes
load class: sun.security.rsa.RSAPrivateKeyImpl
loaded 305 classes
load class: sun.security.pkcs.PKCS8Key
loaded 306 classes
load class: sun.security.rsa.RSAPrivateCrtKeyImpl
loaded 307 classes
load class: java.security.interfaces.RSAPrivateCrtKey
loaded 308 classes
load class: sun.security.rsa.RSACore
loaded 309 classes
load class: sun.security.rsa.RSAPadding
loaded 310 classes
load class: java.util.Collections\$SynchronizedMap
loaded 311 classes
load class: java.util.Base64
loaded 312 classes
load class: java.util.Base64\$Decoder
loaded 313 classes
load class: java.util.Base64\$Encoder
loaded 314 classes
load class: javax.crypto.NullCipher
loaded 315 classes
load class: java.math.MutableBigInteger
loaded 316 classes

Exception in thread "main" load class:
java.lang.Throwable\$WrappedPrintStream
loaded 317 classes
load class: java.lang.Throwable\$PrintStreamOrWriter
loaded 318 classes
javax.crypto.BadPaddingException: Decryption error
load class: java.lang.StackTraceElement\$HashedModules
loaded 319 classes
at
java.base/sun.security.rsa.RSAPadding.unpadV15(RSAPadding.java:
378)
at
java.base/sun.security.rsa.RSAPadding.unpad(RSAPadding.java:290
)
at
java.base/com.sun.crypto.provider.RSACipher.doFinal(RSACipher.j
ava:359)
at
java.base/com.sun.crypto.provider.RSACipher.engineDoFinal(RSAC
ipher.java:392)
at java.base/javax.crypto.Cipher.doFinal(Cipher.java:2205)
at Licence.decryptText(Licence.java:49)
at Licence.main(Licence.java:65)
load class: java.util.IdentityHashMap\$IdentityHashMapIterator
loaded 320 classes
load class: java.util.IdentityHashMap\$KeyIterator
loaded 321 classes
load class: java.lang.Shutdown
loaded 322 classes
load class: java.lang.Shutdown\$Lock
loaded 323 classes

Висновок: у ході лабораторної роботи досліджено можливості java-агентів з використанням мови Java.