Cryptography with OpenSSL

Overview

The purpose of this lab is to practice different encryption algorithms with OpenSSL. You will need to use different hash algorithms and test the performance of AES-256.

RSPEC file:

<https://raw.githubusercontent.com/DrVoyager/EdGENI/master/Rspec-Files/SymmetricEncryptionHash-Rspec.txt>

**Preparation**

**Before you start, you will need to complete with the following setups (If you have completed some steps before, you can skip them):**

1. Setup user account on GENI and join a Project. If you have already registered an account, use your existing account. However, join the project.
2. Setup Lab in GENI. When choosing the RSPEC file, click the “URL” radio button and paste the following url, and click “Select” button. <https://raw.githubusercontent.com/DrVoyager/EdGENI/master/Rspec-Files/SymmetricEncryptionHash-Rspec.txt>
3. Then follow the remaining steps in the 2-Setup-Lab document to select a site and reserve the resource. (If you are not sure what to do, see *2-Setup-Lab*)
4. Establish SSH connections to the GENI virtual machine and obtain the VNC GUI (if you are not sure what to do, see *3-Connect-to-VM*)
5. On the VM, open a terminal, type in “**openssl version**” to verify if OpenSSL has already been installed. In most cases, OpenSSL has been installed on the VM. If it does not, type in the command “**sudo apt-get install openssl**” to install it.
6. If the VM does not have a web browser, you can install one by typing “**sudo apt-get install firefox**”. After installing, you can use firefox by typing the following command on the terminal: “**firefox &**”.

**Note:** If you are using GENI VM to work on this lab, you can upload or download file by using WinSCP. Or you can share file by using google drive via Firefox.

**Task 1 – Get familiar with OpenSSL**

1. Follow the instructions in [https://www.madboa.com/geek/openssl/#how-do-i-create-an-md5-or-sha1-digest-of-a-file (Links to an external site.)Links to an external site.](https://www.madboa.com/geek/openssl/#how-do-i-create-an-md5-or-sha1-digest-of-a-file) to get familiar with how to generate digest (hash value) of a file.
2. Follow the following instruction to learn how to use AES-256 encryption algorithms to encrypt and decrypt a file: [https://www.madboa.com/geek/openssl/#how-do-i-simply-encrypt-a-file (Links to an external site.)Links to an external site.](https://www.madboa.com/geek/openssl/#how-do-i-simply-encrypt-a-file). Especially study how to encrypt/decrypt without typing passphrase (i.e., the usage of the “-pass” parameter).
3. **Show the professor that you are capable of generating Hash and Encrypt/Decrypt file without typing a passphrase.**

**Task 2 – Generating digest for files**

1. Select a project from <http://www.apache.org/index.html#projects-list>. **The first character of the project name should be the same as the first character of your first name.** For instance, if your name is *John Smith*, you can choose project *JMeter*. Click the link of that project, and find the download page. In the download page, you might see several packages are listed. Select the package that have digest link besides it and download the package. For instance, in the following page (Figure 1, the download page of project *Olingo*), you can select “*Olingo OData2 Sources*”, because it has a digest next to it available, i.e., *md5* and *sha512*. Download that package. From the digests provided in the webpage, **select at least two hash algorithms** (in the example case, the two algorithms are *md5* and *sha512*. If a package only lists the hash value of one digest, then you can choose to select that only algorithm). **For each algorithm you choose, use OpenSSL to compute its digest and compare the generated hash value with the one listed on the webpage.** The two hash values should be the same. **Make a screenshot of the package you choose (as Figure 1 shows), your OpenSSL digest execution result (as Figure 2 shows), and webpage showing the correct digest value (as Figure 3 shows). Also, paste the address of the webpage that shows the digest value. Put all the screenshots and the webpage address to your report, giving proper comments describing what hash algorithm you use and what is the length of the hash value**.

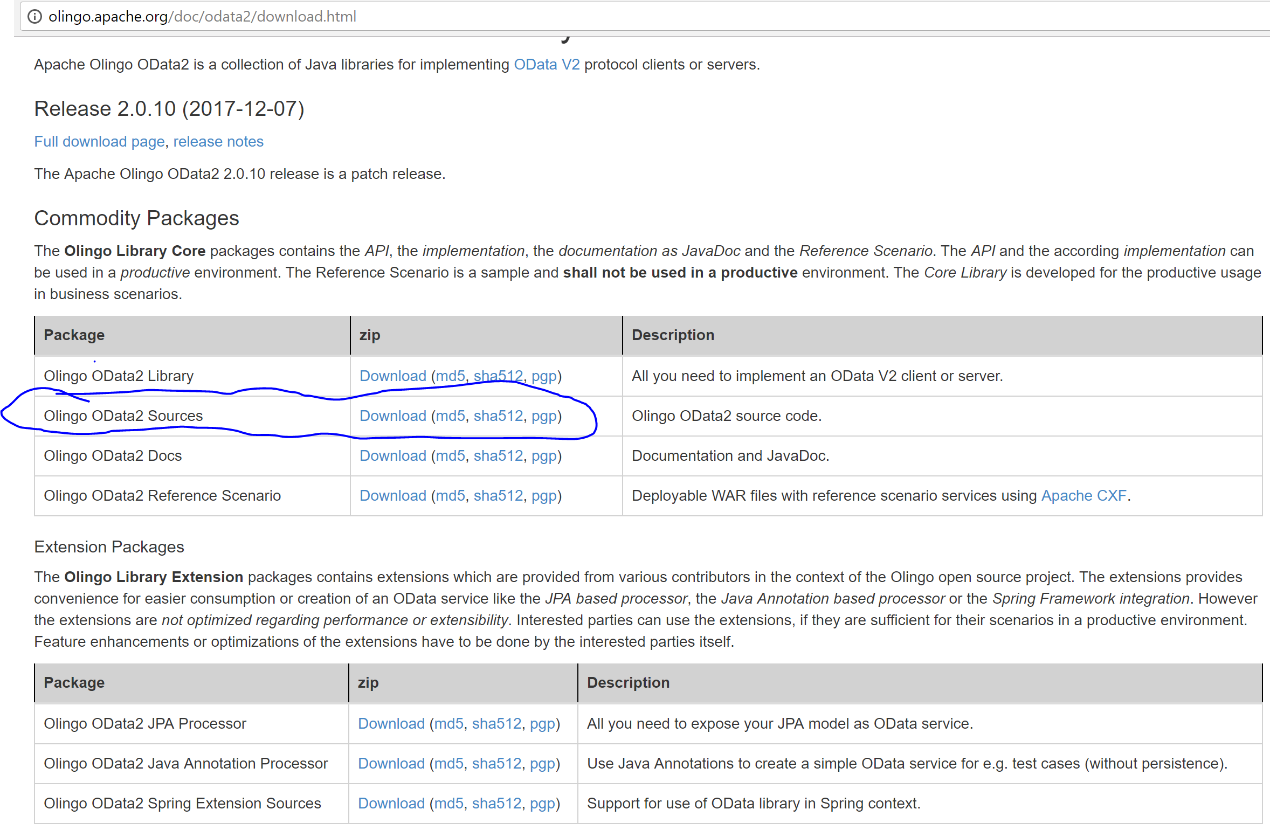


Figure 1 Chosen Package Screenshot

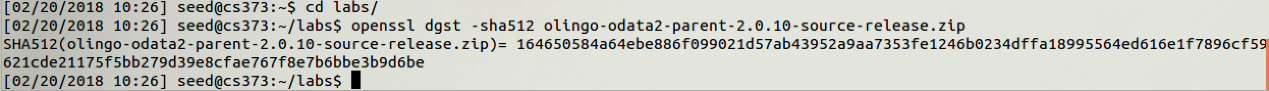


Figure 2 OpenSSL Output Screenshot

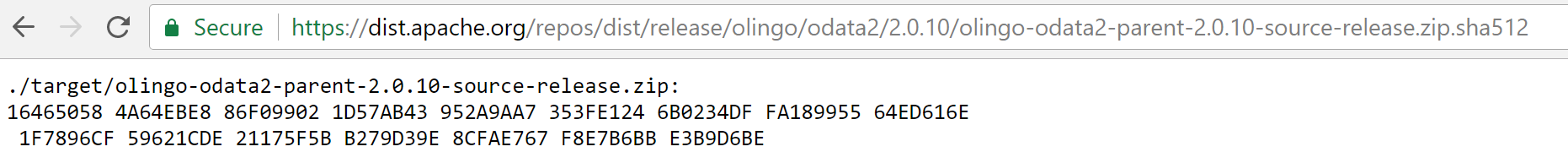


Figure 3 Digest in Website Screenshot

**Task 3 – File encryption and decryption**

1. Prepare three files to encrypt. The size of the first file is between 1KB and 512KB; the size of the second file is between 1MB and 512MB; The size of the third file is above 1GB. You can find a word document, which is usually between 1K and 512K. You can download project packages from apache.org, which are mostly between 1M and 512MB. The virtual machine image file (e.g., the ubuntu virtual machine used for hands-on lab 1) or operating system installation package (e.g., the windows 8 iso file used in assignment 1) is beyond 1 GB.
2. Encrypt and decrypt the three files with OpenSSL using aes-256-cbc mode. Measure the execution time for encryption and decryption.
   * a) Use “time” command in Linux to measure the encryption/decryption time. The usage of “time” can be found in this webpage: [https://www.lifewire.com/command-return-time-command-4054237 (Links to an external site.)Links to an external site.](https://www.lifewire.com/command-return-time-command-4054237).
   * b) Avoid typing a passphrase every time you encrypt or decrypt a file, using “-pass” parameter.
   * c) For the accuracy of the measurement, for the same file, execute it multiple times (at least five times), and find the average value.
   * **d) Based on your evaluation, fill in the following form.**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| File Size Range | File Name | File Size | Encryption Command | Average Encryption Time | Decryption Command | Average Decryption Time |
| 1KB-512KB |  |  |  |  |  |  |
| 1MB-512MB |  |  |  |  |  |  |
| Beyond 1GB |  |  |  |  |  |  |

Notes

We can also finish the work on your won Windows operating system. If you are using a Windows OS, you should download *OpenSSL for Windows* to your machine, unzip it so that you can use it. You can download the “Binaries” in the address [http://gnuwin32.sourceforge.net/packages/openssl.htm (Links to an external site.)Links to an external site.](http://gnuwin32.sourceforge.net/packages/openssl.htm). Then unzip it.

For Task 3.2.a), Windows does not support “time” command. However, you can refer to this method: [https://stackoverflow.com/a/32296674 (Links to an external site.)Links to an external site.](https://stackoverflow.com/a/32296674). It works for most Windows OS.