

Forum:[Hack Forums](#)

Thread: [Any licensing methods for C++?](#)

SuddenStrike2 - July 1, 2024, 4:01 pm

Just wondering if you guys know any method on how to make licensed applications written in C++ for windows application? Something easy as a reference from a trusted forum will do. Sorry if I asked here, its just that sometimes articles are not that reliable nowadays. Hope to get a response. Thanks.

notagh0st - July 2, 2024, 8:27 am

(Jul 1, 2024 04:01 PM)SuddenStrike2 Wrote: [\[CITE\]](#) Just wondering if you guys know any method on how to make licensed applications written in C++ for windows application? Something easy as a reference from a trusted forum will do. Sorry if I asked here, its just that sometimes articles are not that reliable nowadays. Hope to get a response. Thanks.

You need to decide what kind of license you want to provide first. Is this going to be subscription based? Single-use license? And then are you wanting to support offline activation or not? Do you want to allow multiple devices per license or not? etc. If you can be more specific, I can point you in the right direction.

SuddenStrike2 - July 2, 2024, 8:42 pm

(Jul 2, 2024 08:27 AM)notagh0st Wrote: [\[CITE\]](#) You need to decide what kind of license you want to provide first. Is this going to be subscription based? Single-use license? And then are you wanting to support offline activation or not? Do you want to allow multiple devices per license or not? etc. If you can be more specific, I can point you in the right direction.

Thanks for asking clarification. Well since you already have mentioned those, then let's start with a single-use license that can be used offline. I know it would easily be reverse engineered but I just want to know man. Is that good to ask?

notagh0st - July 4, 2024, 9:40 pm

(Jul 2, 2024 08:42 PM)SuddenStrike2 Wrote: [\[CITE\]](#) Thanks for asking clarification. Well since you already have mentioned those, then let's start with a single-use license that can be used offline. I know it would easily be reverse engineered but I just want to know man. Is that good to ask?

Sure thing, here you go. The below example is source code for a C++ program that reads a license key out of a file called license.txt in the same directory as the program. Since you asked for offline activation, the license get that gets generated would need to be something that the program can determine without an internet connection. In this case, I've decided to take the name of the computer on the network, and the username of the logged in user, and create a sha1 hash out of that, and make that the license key. But it could really be anything, there are a lot of programs that create custom encryption/decryption methods where they store expiration date info for the license inside of an RSA key, but I'm keeping it simple for you. I currently have a log statement that shows what the license "should" be when you run it. This is just for debugging purposes, so you can know how the hash is generated, and what should be put in the license.txt file.

Here's the code:

```
Code #include <iostream>
#include <fstream>
#include <sstream>
```

```

#include <string>
#include <openssl/sha.h>
#include <unistd.h>
#include <sys/types.h>
#include <pwd.h>
#include <sys/utsname.h>

// Function to read the content of license.txt
std::string readLicenseFile(const std::string& filename) {
    std::ifstream file(filename);
    if (!file) {
        std::cerr << "Unable to open file " << filename << std::endl;
        return "";
    }
    std::stringstream buffer;
    buffer << file.rdbuf();
    return buffer.str();
}

// Function to get the network name of the computer
std::string getNetworkName() {
    struct utsname buffer;
    if (uname(&buffer) != 0) {
        std::cerr << "Failed to get network name" << std::endl;
        return "";
    }
    return std::string(buffer.nodename);
}

// Function to get the logged-in user's username
std::string getUsername() {
    struct passwd* pw;
    uid_t uid = geteuid();
    pw = getpwuid(uid);
    if (pw) {
        return std::string(pw->pw_name);
    }
    std::cerr << "Failed to get username" << std::endl;
    return "";
}

// Function to generate SHA-1 hash of a string
std::string generateSHA1(const std::string& input) {
    unsigned char hash[SHA_DIGEST_LENGTH];
    SHA1(reinterpret_cast<const unsigned char*>(input.c_str()), input.size(), hash);
    std::stringstream ss;
    for (int i = 0; i < SHA_DIGEST_LENGTH; ++i) {
        ss << std::hex << std::setw(2) << std::setfill('0') << (int)hash[i];
    }
    std::cout << "License should be: " << ss.str() << std::endl;
    return ss.str();
}

```

```
// Function to verify the license
bool verifyLicense(const std::string& licenseHash) {
    std::string networkName = getNetworkName();
    std::string username = getUsername();
    if (networkName.empty() || username.empty()) {
        return false;
    }

    std::string combined = networkName + username;
    std::string computedHash = generateSHA1(combined);

    return (computedHash == licenseHash);
}

int main() {
    std::string licenseContent = readLicenseFile("license.txt");
    if (licenseContent.empty()) {
        std::cerr << "License file is empty." << std::endl;
    }

    if (verifyLicense(licenseContent)) {
        // continue executing program here
        std::cout << "Hello world" << std::endl;
    } else {
        std::cerr << "License verification failed." << std::endl;
    }

    return 0;
}
```

Build it with the -lssl -lcrypto flags. I.e. for me, I built it using: `g++ main.cpp -o verify_license -lssl -lcrypto`

Then run it with `./verify_license`.

You'll see something like this the first time you run it without a license:

```
./verify_license
License should be: 02ddc951352ebdd578309445c85fa44fd6bab33a
License verification failed.
```

In my test case, the network name is testNetwork and the username is testUser, combined together, the sha1 hash is 02ddc951352ebdd578309445c85fa44fd6bab33a so after I put 02ddc951352ebdd578309445c85fa44fd6bab33a in my license.txt file, I get:

```
./verify_license
License should be: 02ddc951352ebdd578309445c85fa44fd6bab33a
Hello world
```

The idea being that where the "Hello world" log in the code is, you'd continue executing whatever program you're wanting to protect.

secret_code - July 5, 2024, 1:14 am

(Jul 4, 2024 09:40 PM)notaght0st Wrote: [CITE](#) Sure thing, here you go. The below example is

source code for a C++ program that reads a license key out of a file called license.txt in the same directory as the program. Since you asked for offline activation, the license get that gets generated would need to be something that the program can determine without an internet connection. In this case, I've decided to take the name of the computer on the network, and the username of the logged in user, and create a sha1 hash out of that, and make that the license key. But it could really be anything, there are a lot of programs that create custom encryption/decryption methods where they store expiration date info for the license inside of an RSA key, but I'm keeping it simple for you. I currently have a log statement that shows what the license "should" be when you run it. This is just for debugging purposes, so you can know how the hash is generated, and what should be put in the license.txt file.

Here's the code:

```
Code #include <iostream>
#include <fstream>
#include <sstream>
#include <string>
#include <openssl/sha.h>
#include <unistd.h>
#include <sys/types.h>
#include <pwd.h>
#include <sys/utsname.h>

// Function to read the content of license.txt
std::string readLicenseFile(const std::string& filename) {
    std::ifstream file(filename);
    if (!file) {
        std::cerr << "Unable to open file " << filename << std::endl;
        return "";
    }
    std::stringstream buffer;
    buffer << file.rdbuf();
    return buffer.str();
}

// Function to get the network name of the computer
std::string getNetworkName() {
    struct utsname buffer;
    if (uname(&buffer) != 0) {
        std::cerr << "Failed to get network name" << std::endl;
        return "";
    }
    return std::string(buffer.nodename);
}

// Function to get the logged-in user's username
std::string getUsername() {
    struct passwd* pw;
    uid_t uid = geteuid();
    pw = getpwuid(uid);
    if (pw) {
        return std::string(pw->pw_name);
    }
}
```

```

}
std::cerr << "Failed to get username" << std::endl;
return "";
}

```

```

// Function to generate SHA-1 hash of a string
std::string generateSHA1(const std::string& input) {
    unsigned char hash[SHA_DIGEST_LENGTH];
    SHA1(reinterpret_cast<const unsigned char*>(input.c_str()), input.size(), hash);
    std::stringstream ss;
    for (int i = 0; i < SHA_DIGEST_LENGTH; ++i) {
        ss << std::hex << std::setw(2) << std::setfill('0') << (int)hash[i];
    }
    std::cout << "License should be: " << ss.str() << std::endl;
    return ss.str();
}

```

```

// Function to verify the license
bool verifyLicense(const std::string& licenseHash) {
    std::string networkName = getNetworkName();
    std::string username = getUsername();
    if (networkName.empty() || username.empty()) {
        return false;
    }

    std::string combined = networkName + username;
    std::string computedHash = generateSHA1(combined);

    return (computedHash == licenseHash);
}

```

```

int main() {
    std::string licenseContent = readLicenseFile("license.txt");
    if (licenseContent.empty()) {
        std::cerr << "License file is empty." << std::endl;
    }

    if (verifyLicense(licenseContent)) {
        // continue executing program here
        std::cout << "Hello world" << std::endl;
    } else {
        std::cerr << "License verification failed." << std::endl;
    }

    return 0;
}

```

Build it with the -lssl -lcrypto flags. I.e. for me, I built it using: g++ main.cpp -o verify_license -lssl -lcrypto

Then run it with ./verify_license.

You'll see something like this the first time you run it without a license:

```
./verify_license
```

```
License should be: 02ddc951352ebdd578309445c85fa44fd6bab33a
```

```
License verification failed.
```

In my test case, the network name is testNetwork and the username is testUser, combined together, the sha1 hash is 02ddc951352ebdd578309445c85fa44fd6bab33a so after I put 02ddc951352ebdd578309445c85fa44fd6bab33a in my license.txt file, I get:

```
./verify_license
```

```
License should be: 02ddc951352ebdd578309445c85fa44fd6bab33a
```

```
Hello world
```

The idea being that where the "Hello world" log in the code is, you'd continue executing whatever program you're wanting to protect.

nice job, i was wandering tho why not "string_view" function argument type instead of "const string&" ? also where string is ok, "const" is missing, why r u not using const where possible? Another point is, same within "const" context, why not mark function signature return type as const as well? like "bool verifyLicense()" could be "const bool verifyLicense()", and the same applies to other functions as well... Another point is how you initialize variables, why don't u use direct initialization(std::string x("john doe")) techniques instead of copy initialization(std::string x = "john doe")? use direct instead of copy, is less consuming and has less operations to execute. Another point is targeted directly to the "getUsername()" function, you do realize it could just be reduced to 2 line of code and can and "should" also be marked as const? im not trying to sound superior or arrogant :) , i think maybe you would like to see your errors and/or where you can improve your code, that's exactly what happened to me as well when i was at the beginning....

notagh0st - July 5, 2024, 2:42 am

(Jul 5, 2024 01:14 AM)secret_code Wrote: [\[CITE\]](#) nice job, i was wandering tho why not "string_view" function argument type instead of "const string&" ? also where string is ok, "const" is missing, why r u not using const where possible? Another point is, same within "const" context, why not mark function signature return type as const as well? like "bool verifyLicense()" could be "const bool verifyLicense()", and the same applies to other functions as well... Another point is how you initialize variables, why don't u use direct initialization(std::string x("john doe")) techniques instead of copy initialization(std::string x = "john doe")? use direct instead of copy, is less consuming and has less operations to execute. Another point is targeted directly to the "getUsername()" function, you do realize it could just be reduced to 2 line of code and can and "should" also be marked as const? im not trying to sound superior or arrogant :) , i think maybe you would like to see your errors and/or where you can improve your code, that's exactly what happened to me as well when i was at the beginning....

I actually don't know C++. I just gathered this stuff together by reading some stuff online. The goal was to show a pattern or workflow for OP, not to provide an actual solution that they would end up using, just to show how it works.

secret_code - July 5, 2024, 4:05 am

(Jul 5, 2024 02:42 AM)notagh0st Wrote: [\[CITE\]](#) I actually don't know C++. I just gathered this stuff together by reading some stuff online. The goal was to show a pattern or workflow for OP, not to provide an actual solution that they would end up using, just to show how it works.

yeah i guessed, just wanted to make sure someone doesn't make the mistake of using that specific code for real :) as it looks like it was generated by AI instead(reasons: lack of error control flow at least in the main() branch it was kind of good to have, now... this was ok without control flow handling but keep in mind that this is good practice anyway so you should get used to use it whenever you can("just good practice :"). Then the lack of consideration of the current line before proceeding to the next line(.e.g: like reducing getUsername() to 2 lines of code, lack of proper logic

for the entire execution flow(.e.g: in c++ we just return if possible instead of implying "else" statements where possible to reduce complexity as c++ is enough complex by itself). Im being strictly semantic here because it is useful for someone else who didn't know this(in case wants to know :)).

thank you for understanding :)

SuddenStrike2 - July 5, 2024, 9:05 am

(Jul 5, 2024 02:42 AM)notagh0st Wrote: [\[CITE\]](#) I actually don't know C++. I just gathered this stuff together by reading some stuff online. The goal was to show a pattern or workflow for OP, not to provide an actual solution that they would end up using, just to show how it works.

I didn't knew you would do that just for me friend but great thanks for sharing your insight and also for the effort into putting up code. Very much appreciate it!

Klonopin - July 18, 2024, 7:13 am

Consider developing your own API and using web calls or sockets for authentication and licensing. It's advisable to keep as much of the logic as possible server-side and avoid relying on third-party authentication services like KeyAuth for better security and control.

SuddenStrike2 - July 22, 2024, 4:36 pm

(Jul 18, 2024 07:13 AM)Klonopin Wrote: [\[CITE\]](#) Consider developing your own API and using web calls or sockets for authentication and licensing. It's advisable to keep as much of the logic as possible server-side and avoid relying on third-party authentication services like KeyAuth for better security and control.

KeyAuth looks good but sadly its subscription based. Are there any free options (not free trial please) you happen to know or maybe a good alternative perhaps?

Klonopin - July 23, 2024, 4:05 pm

(Jul 22, 2024 04:36 PM)SuddenStrike2 Wrote: [\[CITE\]](#) KeyAuth looks good but sadly its subscription based. Are there any free options (not free trial please) you happen to know or maybe a good alternative perhaps?

I do not recommend keyauth, check out auth0 its free upto like 7k users and its much much more secure.

SuddenStrike2 - July 27, 2024, 1:44 am

(Jul 23, 2024 04:05 PM)Klonopin Wrote: [\[CITE\]](#) I do not recommend keyauth, check out auth0 its free upto like 7k users and its much much more secure.

Yes I have seen Auth0 and I have been reading its documentation. I also checked out its pricing, it somewhat similar to an API called Mapbox when it comes to payment terms.

Klonopin - July 27, 2024, 7:39 am

(Jul 27, 2024 01:44 AM)SuddenStrike2 Wrote: [\[CITE\]](#) Yes I have seen Auth0 and I have been reading its documentation. I also checked out its pricing, it somewhat similar to an API called Mapbox when it comes to payment terms.

Try to keep your application mostly serverside to avoid any cracks from happening.

SuddenStrike2 - July 27, 2024, 8:19 am

(Jul 27, 2024 07:39 AM)Klonopin Wrote: [\[CITE\]](#) Try to keep your application mostly serverside to avoid any cracks from happening.

Okay, thanks for the advice and I will try the implementation also of games in which when they put license, they will download updates to make things work.