

ARKAKAPI

Bimonthly Cyber Security Magazine

06

July-August

www.arkakapimag.com

Copyright Projects Based on Blockchain

The Ship That Got Hacked Ashore

Make your Own Messaging Application with Signal

Client-side Static Analysis in Web Applications

Our Fellow Confidant, RSA

Prophecies for the Next 30 Years of Cyber Security



ISSN 2645-906X



9 772645 906009

Editor's Note

Hey there!

As we are approaching the very first year anniversary of the magazine, on behalf of the Arka Kapi Magazine team, I would like to thank you all, dear readers, for your support. Among the pages of this issue, you once again are going to find amazing articles, technically and finely detailed. From developing your own messaging app to encryption algorithms, blockchain copyrights and sociological implementations about the future of technology, interesting knowledge is waiting to find a place in your minds.

In addition to being informative, another aim of us is to inspire people to discover, learn, and build for the better. Even in today's circumstances where technology advances faster each second, one might think that a new change found is already left in the past. There actually are no competitors; only scientists dedicated

to discovery. For this reason, one should never give up and continue their passion no matter what. I would like to share a quote by the famous mathematician Isaac Newton, who is unquestioningly a legend. A little before his passing, the following quote had been remarked by Sir Newton:

"I do not know what I may appear to the world, but to myself I seem to have been only like a boy playing on the seashore, and diverting myself in now and then finding a smoother pebble or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me."

We would like to thank Netsparker Ltd. for sponsoring this issue!

Cansu Topukçu
editor@arkakapimag.com

ARKAKAPI MAG

Cyber Security Magazine YEAR: 1 – JULY-AUGUST ISSUE: 6 Bimonthly - ISSN: 2645-906X www.arkakapimag.com

Editor in Chief: Ziyahan Albeniz • ziyahan@arkakapimag.com

Editorial Operations Manager: Cansu Topukçu • cansu@arkakapimag.com

Chief Business Officer: Oğuz Aydınılmaz • oguz@arkakapimag.com

Director of Web: Ömer Çıtak • omer@arkakapimag.com

Legal Advisor: Mehmet Pehlivan • mehmet@arkakapimag.com

Assistant research editor: Ayşenur Burak • nurayse47@gmail.com

Translators: Serdar Savaş, Atalay Keleştemur, Hakan Özer

Social Media Directors: Nuri Çilengir, Tayfur Özkara

Social Media:



/arkakapimag



/arkakapimag



/arkakapimag



We are proud to secure all our emails with Tutanota.

CONTENT

CYBER SECURITY CONFERENCES - Ayşenur Burak	4
COPYRIGHT PROJECTS BASED ON BLOCKCHAIN TECHNOLOGY - Mihraç Cerrahoğlu	6
MAKE YOUR OWN MESSAGING APPLICATION WITH SIGNAL - Murat Şişman	8
GREAT THREAT IN EDUROAM ACADEMIC NETWORKS - Besim Altınok	15
THE SHIP THAT GOT HACKED ASHORE - Esref Erol	20
CLIENT-SIDE STATIC ANALYSIS IN WEB APPLICATIONS - Mithat Gogebakan	40
ENCRYPT YOUR DISKS WITH BITLOCKER - Arka Kapı	45
OLDEST OF THE HACKERS II - RICHARD GREENBLATT - Cansu Topukçu	49
OUR FELLOW CONFIDANT, RSA - Bayram Gök	53
PROPHECIES FOR THE NEXT 30 YEARS OF CYBERSECURITY - Utku Şen	65

netsparker

Web Application Security Scanner

Use Netsparker to Identify Exploitable Vulnerabilities and Other Security Flaws in Your Websites, Web Applications & Web Services Before Hackers Do.

Netsparker scanners employ the unique, dead accurate & fast **Proof-Based Vulnerability Scanning Technology** that automatically verifies the identified vulnerabilities with a proof of exploit, so you do not have to manually verify them.



Trusted by

 ERNST & YOUNG
Quality In Everything We Do




 SAMSUNG



 ISACA
Institute of Information Systems Auditors

 Microsoft

 ING 

 Booz | Allen | Hamilton

 SIEMENS

Cyber Security Conferences



THE CYBSEC AND BLOCKCHAIN HEALTH

July 11-12, 2019
London, United Kingdom

The importance of cyber security in health sector, ensuring safety in medical devices and technological solutions will be discussed.

Info: <https://bit.ly/31V37yx>

INTERFACE-KANSAS CITY 2019

July 18, 2019
Overland Park Convention Center, USA

Covering IT Infrastructure, BC/DR, IT Security, and Enterprise Communications, you'll find presentations, panel discussions and exhibitors covering a variety of topics, as well as the latest innovations and best practices.

Info: <https://f2fevents.com/event/kcm19/>



CYBERTECH MIDWEST 2019

July 24-25, 2019
Indianapolis, USA

This conference includes all the latest technological innovations, threats and solutions to combat threats in the global cyber arena.

Info: <https://bit.ly/2QM2GQw>



INTERNATIONAL CONFERENCE ON INTERNET MONITORING AND PROTECTION (ICIMP)

July 28-August 02, 2019
Nice, France

This conference focuses on Trends on monitoring with new technologies, Internet traffic surveillance and interception, Internet performance, Security for Internet-based real-time systems, Disaster prevention and recovery, Networks and applications emergency services, User safety, privacy, vulnerabilities, and etc.

Info: <https://bit.ly/2KMK17c>



PYCON AU 2019

Aug 02-06, 2019

ICC Sydney, Sydney, Australia

PyCon Australia ("PyCon AU") is the national conference for the Python Programming Community, bringing together professional, student and enthusiast developers with a love for developing with Python.

Info: <https://2019.pycon-au.org/>

**BLACK HAT USA**

Aug 03-08, 2019

Mandalay Bay, Las Vegas, Nevada, United States

Black Hat is the most technical and relevant information security event series in the world. After four days of information security trainings, there will be a conference.

Info: <https://www.blackhat.com/us-19/>

CLOUD NATIVE SERVICE TRANSFORMATION BERLIN

Aug 17-19, 2019

Hotel Novotel Berlin Mitte, Berlin, Germany

This event is a technology transformation event aggregating Technology Leaders, Industry Experts, Service Providers, Enterprises, Entrepreneurs & Startups alongside with Business Leaders.

Info: <https://bit.ly/2CQRii3>

**GARTNER SECURITY & RISK MANAGEMENT SUMMIT - MUMBAI**

Aug 26-27, 2019

Renaissance Mumbai Convention Centre Hotel, Mumbai, India

This event includes cybersecurity, risk management and compliance strategies, agile architectures, application and data security, cloud and emerging technology security, OT security and much more.

Info: <https://gtnr.it/2LQf3ZA>



Copyright Projects Based On Blockchain Technology

Blockchain Projects developed to protect copyrights and payment methods

Nowadays along with the spreading of the internet, content sharing and consumption increase with the same speed. This has accelerated social development and access to content as well as ease. From the consumer's perspective, this easiness can be a nightmare. Copying, sharing and misusing an article, an artwork or a digital good is very popular nowadays. Although Youtube takes precautions to protect music and video rights, the fairness of the payment policy relative to watch rates is very controversial. Moreover, even if there weren't central systems, you are forced to trust a company about product security. Though it may not seem like a big problem for now, with the introduction of the blockchain technology in every field, discussing this field would become quite usual. Alright, is there a community that focuses on this problem and develops solutions for this? Of course, there are, so, let's look at some of them.

Peertrack: Musical works service like Spotify or Audio Jungle. However, you can payments can be made using cryptocurrency directly - no need to give your card details! Pros are that this service guarantees the work creators (authors) 95% of the fee. So it deserves applauses when considered that Audio Jungle gives between 50-70% to the creator!

Copytrack: They guarantee copyright protection in 140 countries and state that there is no financial risk (taxa-

tion etc.). However, a commission is only taken at the sale, and process immediately. This is possible thanks to Smart Contract technologies - thus you don't depend on companies. They are already processed on the stock exchange and if no deviation takes place in their roadmap, they will start accepting users globally in the second quarter of this year.

Binded: Formerly named as Blockai, Binded is a system designed rather for reading articles and other writing types. As soon as an author uploads his/her article, a record is created with a timestamp on the blockchain. The work owner is informed in case if any violation like copying/stealing happens!

There are so many projects that run on this theme, however, for the sake of a reasonable length of the article, I am going to state only 3 of them:

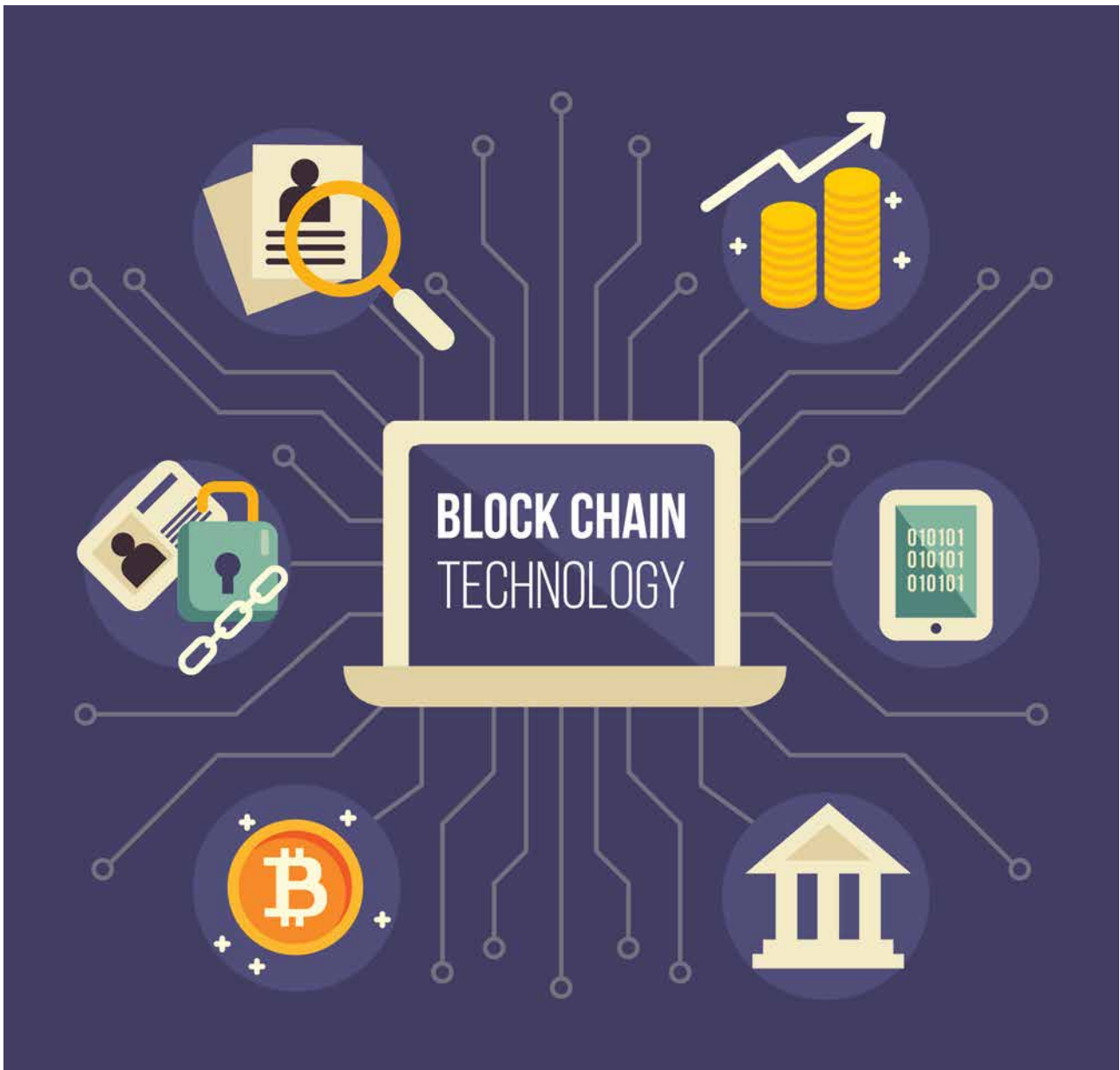
- Mediachain Lab
- Ascribe
- Ujo Music

What do these projects offer the user or content creators globally?

Let's take a look at an example from Turkey. Those who sell their works on services like Envato or Istockphoto have been struggling ever since when PayPal ended its operations in Turkey. There are difficulties for both the creator and buyer! All producers have started to struggle with banks or other payment methods: money given to the intermediary, difficulty of

formal transactions, the paperwork... -and also the exchange rates! When I looked at the projects above, I have not seen assertive ones about protecting the works. Videohive(envanto) can as well handle situations like protection by law or hiding the production date. Here, the trustability of the company is open to arguments. If enslavement to monopoly is what your concern is, we can say that it is possible to say that the works are not stored on the blockchain. So, you

are partially on a central file server and database. If I did not oversee any details. What the hopeful thing is that with the arrival of the above-mentioned payment methods and the prevention of copying with projects like Binded, all this suffering will end. In addition, I find all blockchain projects promising. I even started learning Solidity to develop Smart Contracts to realize the promising projects I have in mind - not to trick people using a Coin!

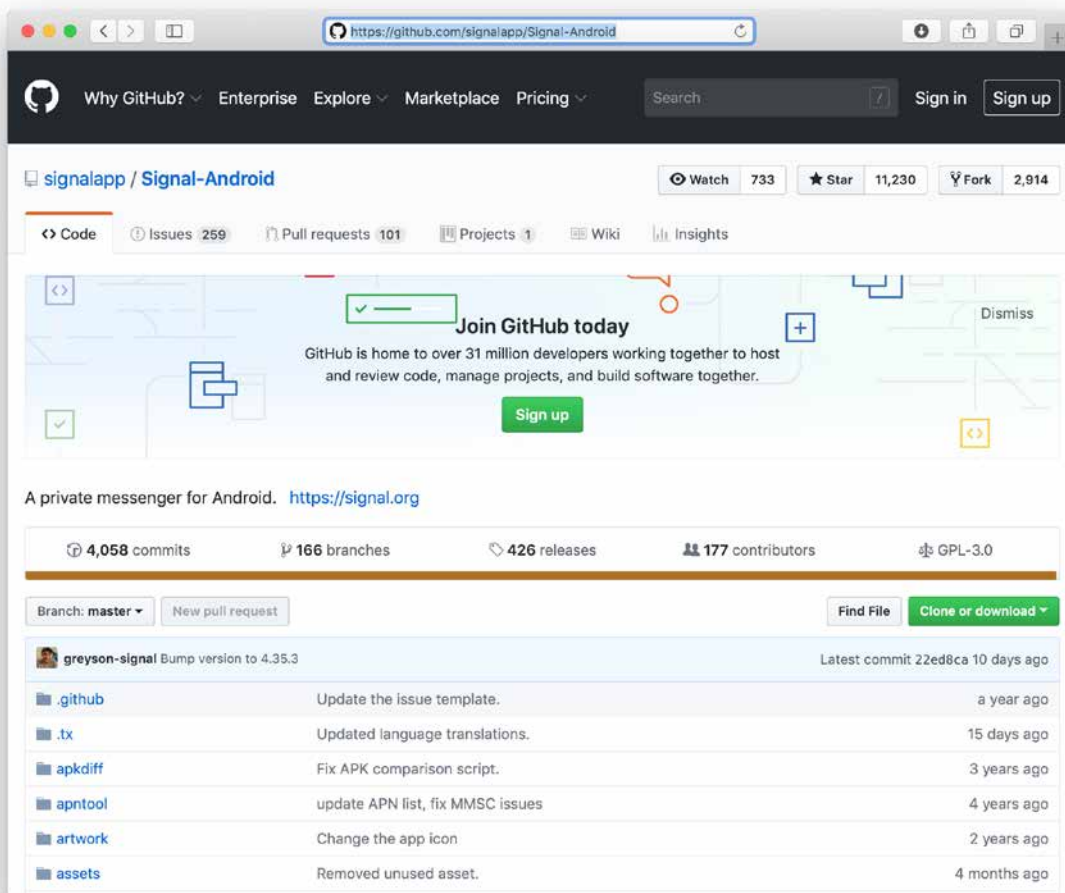


Make your own messaging application with Signal

In recent days, security questions in messaging applications are on the agenda. In the previous issues of the Arka Kapı Magazine, some articles described which messaging applications use which security methods in detail, but this time we will create our own secure messaging application.

Signal

An open source and a system distributed for free which offers a secure messaging environment with end-to-end encryption. There are pre-prepared application codes for devices like Android and iOS are also published on Github, and anyone can download these codes to their computers, make the necessary changes and publish them under other names.

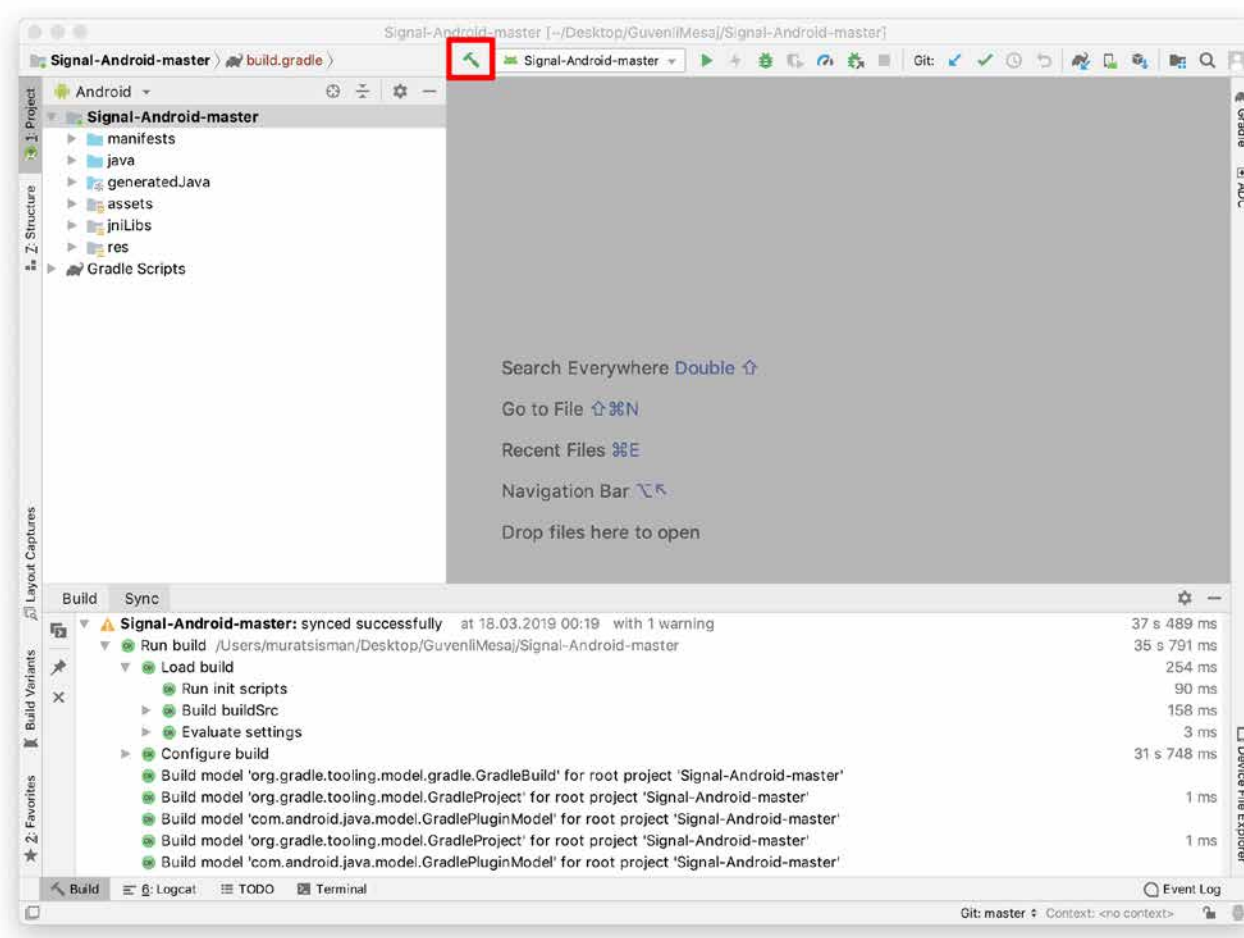


(Signal iOS and Android app source codes are available at <https://github.com/signalapp>)

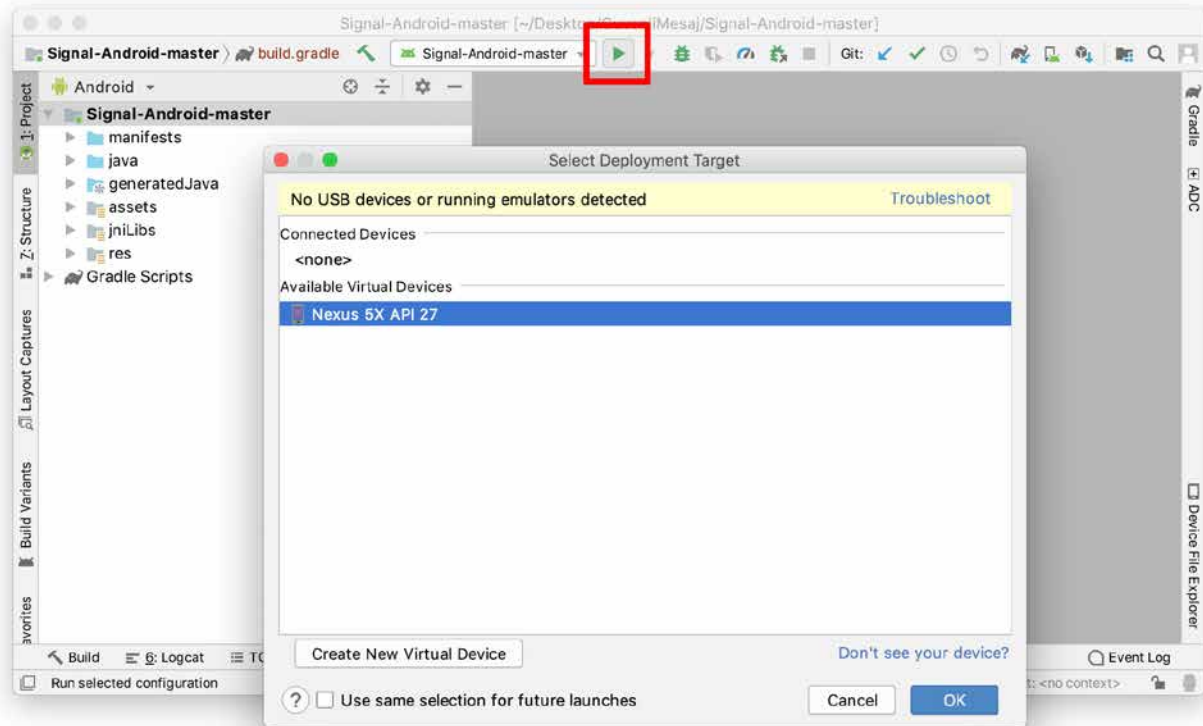
We will create the Android version of the messaging application in our article, and our readers can download the iOS version from the same Github account.

Secure Message V1.0

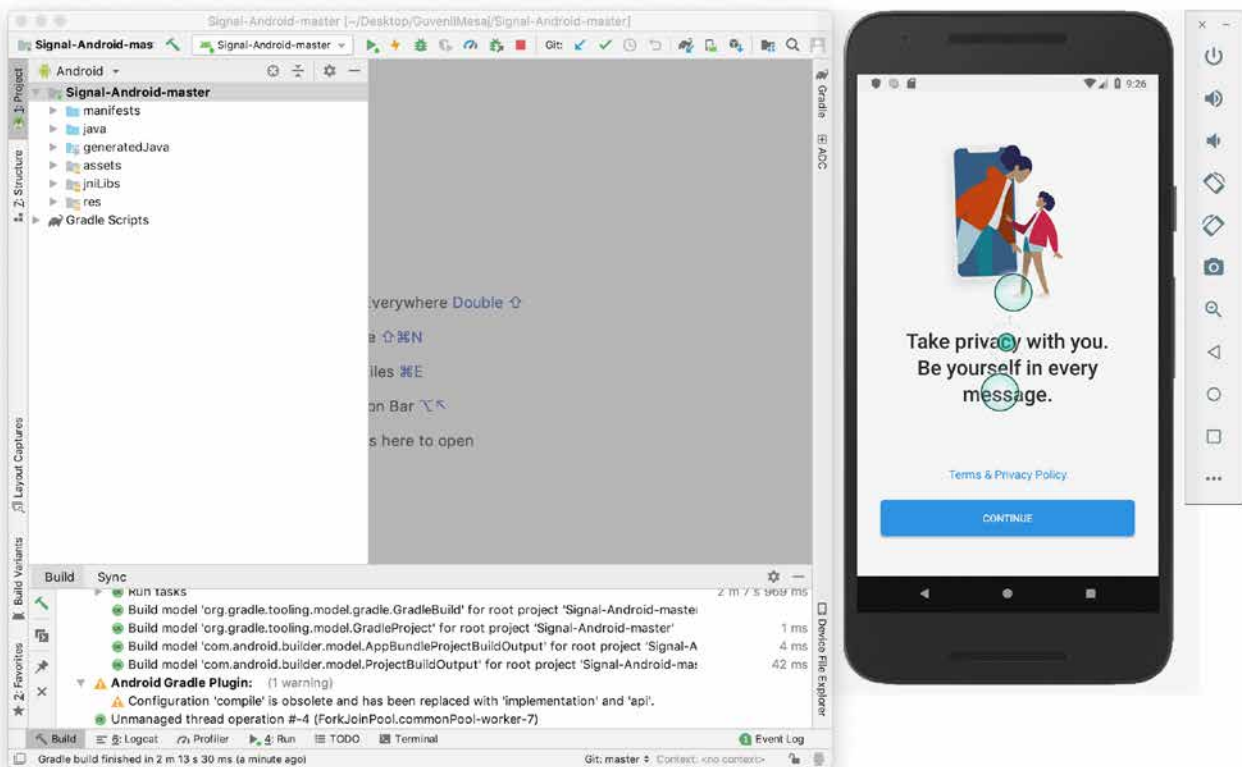
We need to open the codes for the Android application that we downloaded from the Github account with Android Studio software, then click on the hammer icon and complete Gradle Build process. After successful completion, the Signal-android-master module will be visible, and when you click on the run button, it will be able to test from the Virtual Device (simulator) or the device itself.

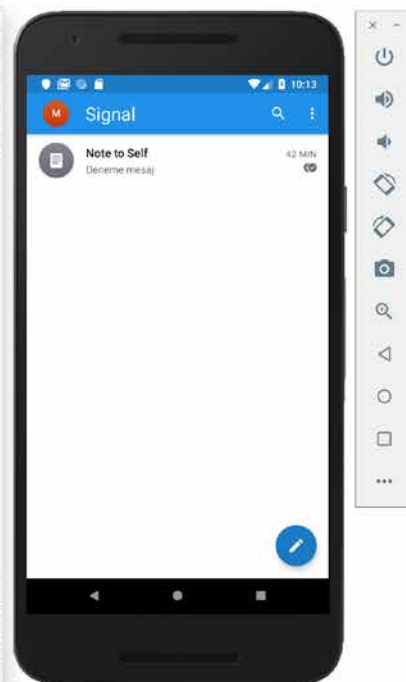
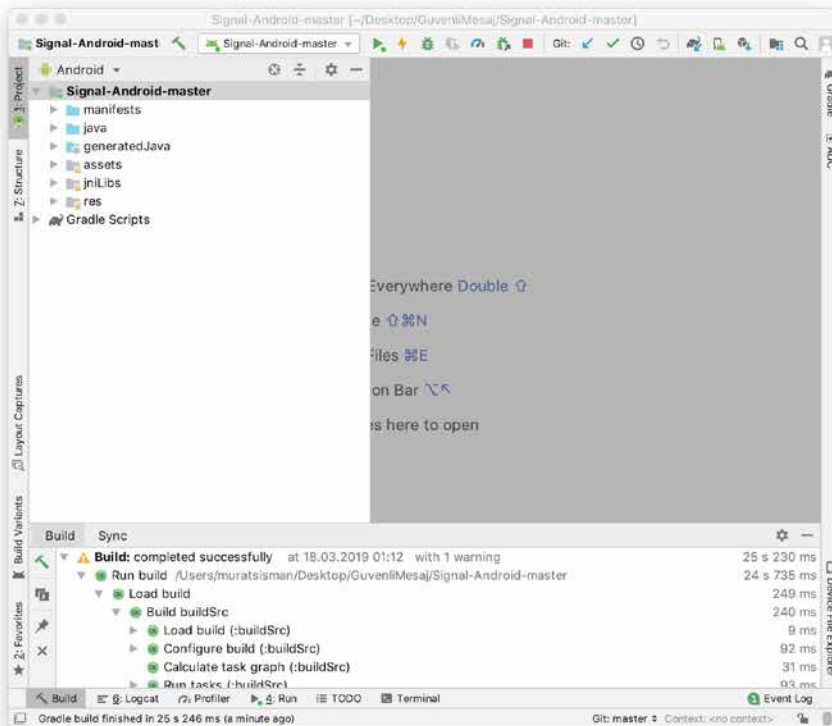
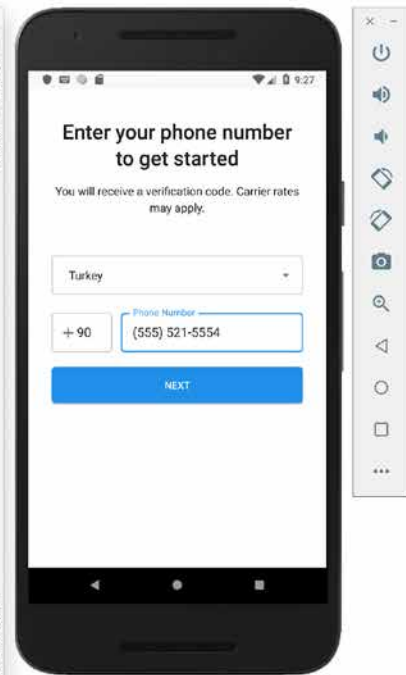
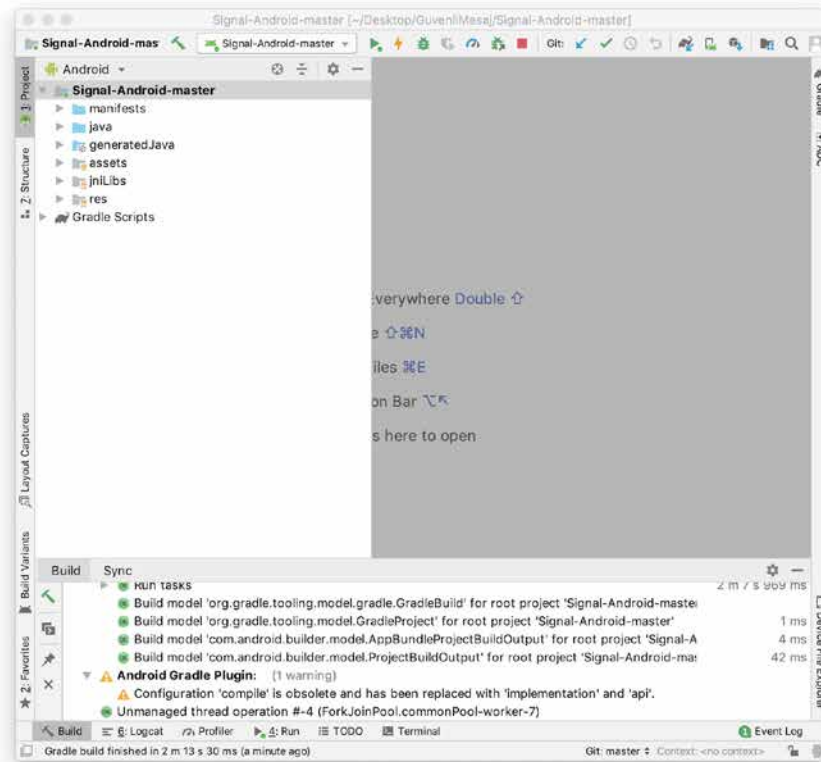


When we press the Run button in Android Studio, a screen appears for selecting the device which the application will run on. If your device is connected to the computer via a USB cable, it will appear in this list, or you can create a new Virtual Device (simulator) to run it.



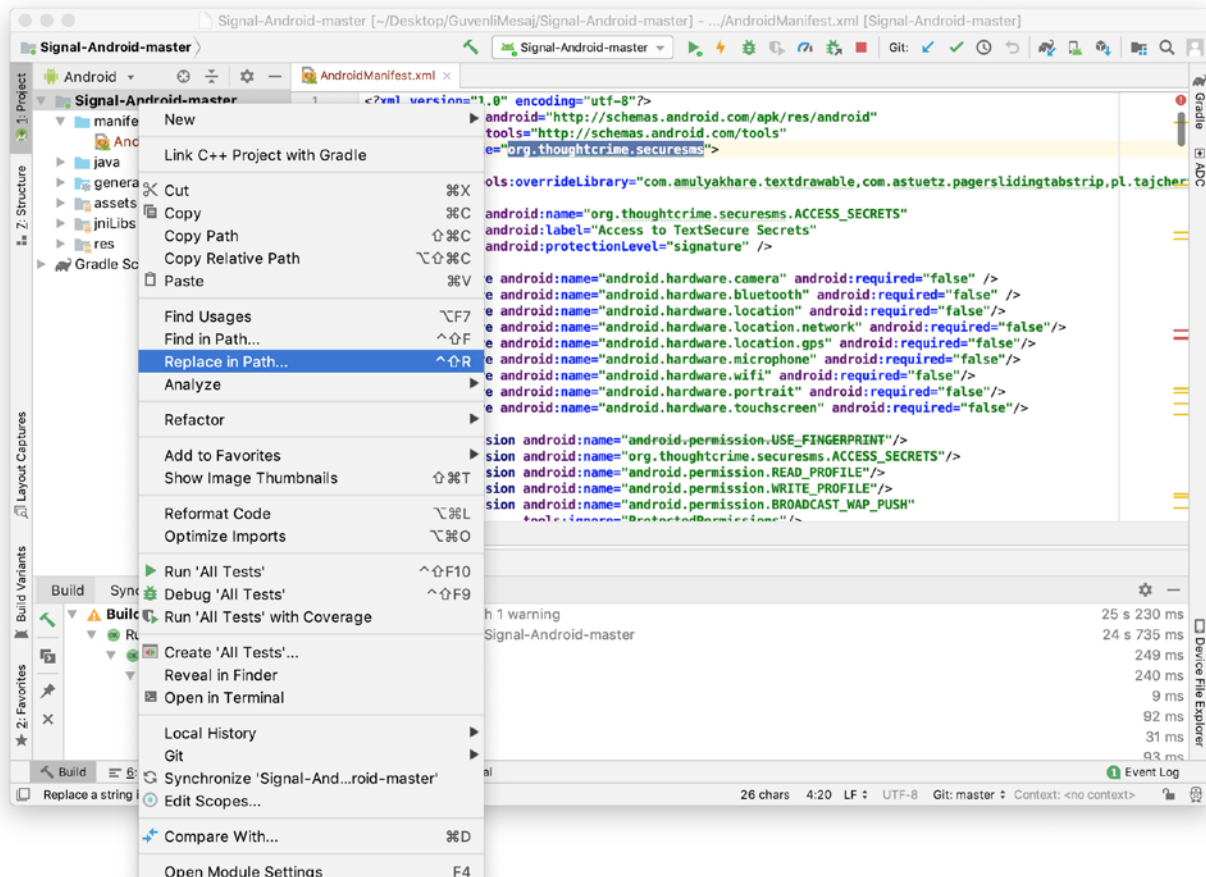
Here is the Signal app!



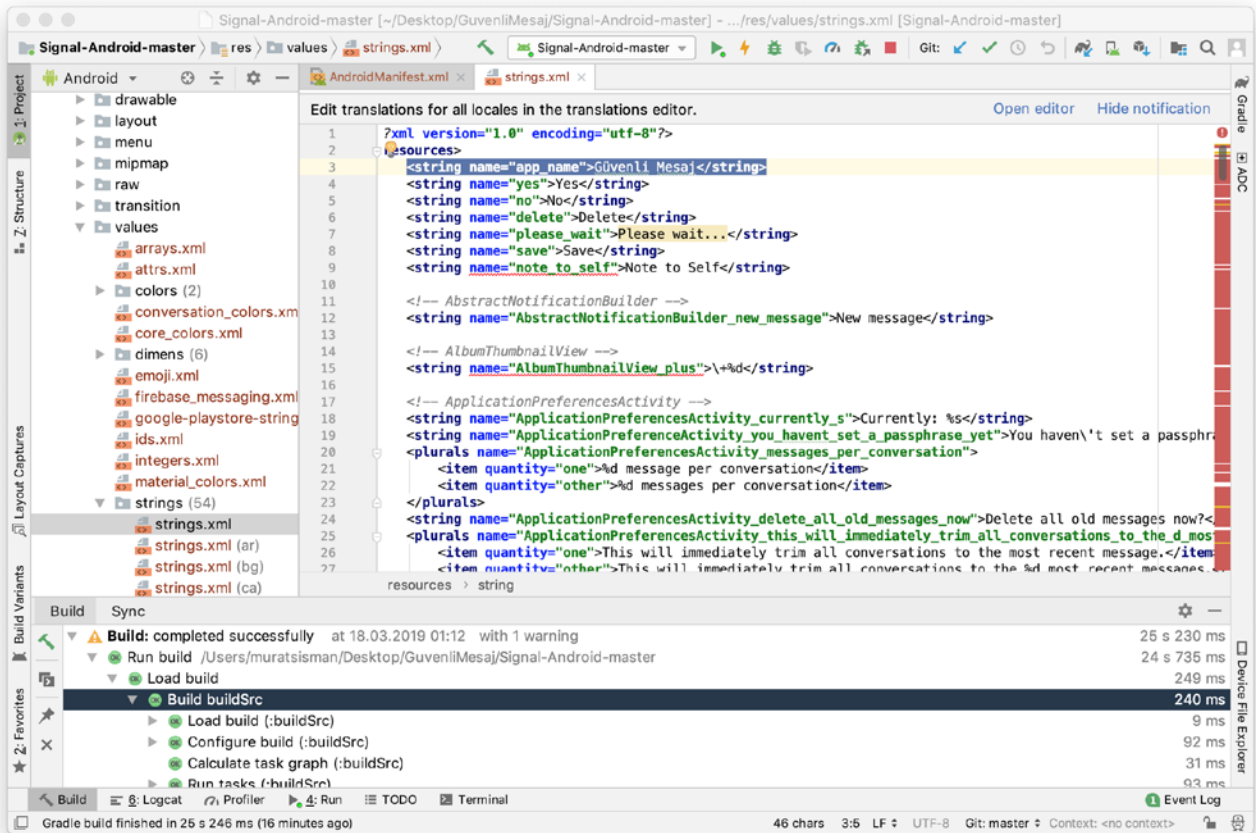


We have compiled the original version of the Signal application and ran it in the simulator. Now, we need to translate these codes into our own application.

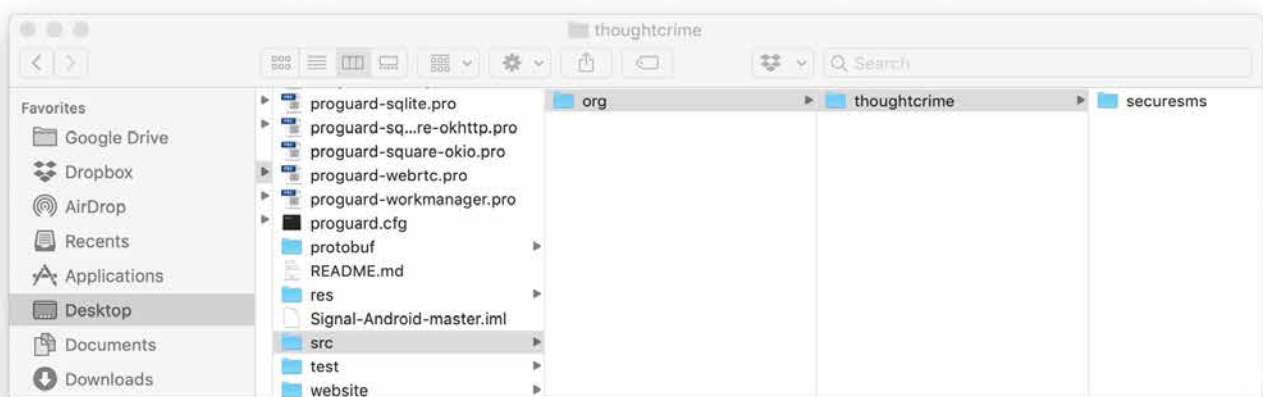
At the first row in the *AndroidManifest.xml* file, “org.thoughtcrime.securesms” string value which refers to the name of the signal application should be replaced with your messaging application name. In our example, we put “com.muratsisman.guvenlimesaj” as the application name. The shortest way for replacing all string values; please click the project and use *Replace in Path* function.



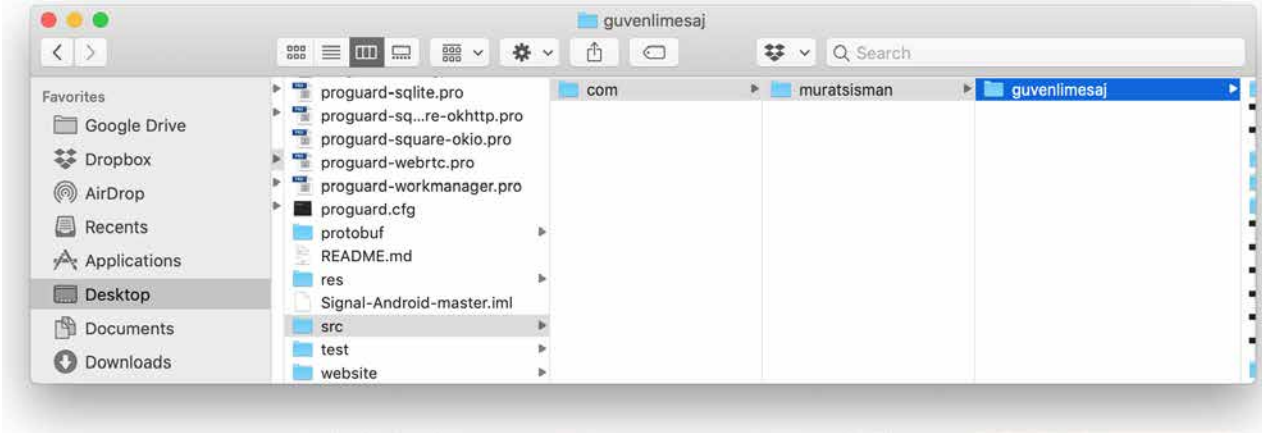
As a final step, we change the `app_name` field in the `strings.xml` file and then the names in the folder containing the source code as “com.muratsisman.guvenlimesaj” for replacing the name in the application screen.



(app_name field in strings.xml)

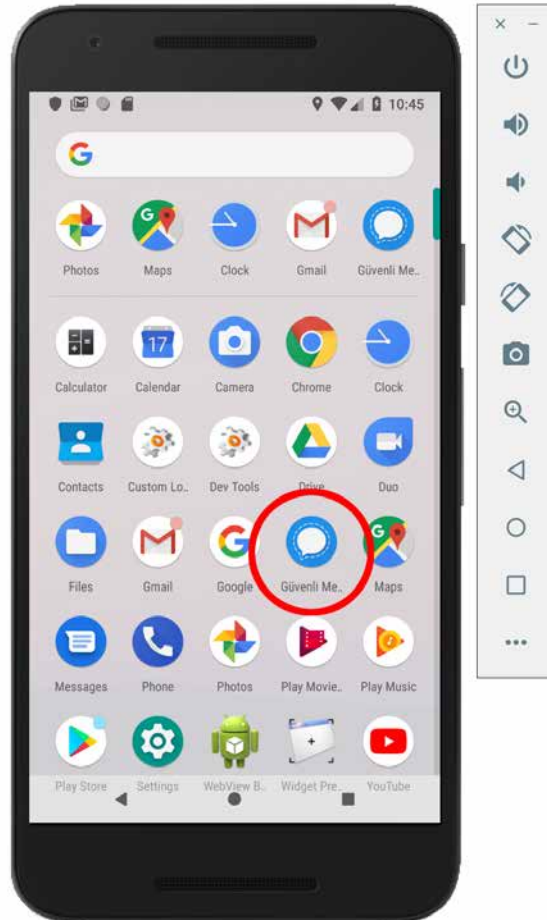


(src -> org -> thoughtcrime -> securesms)



(src -> com -> muratsisman -> guvenlimesaj)

Our application has been prepared completely different from the original Signal under the name of “Güvenli Mesaj”. You can also publish your own secure messaging application in the GooglePlay store by changing the logo and visual elements in the content.



Great Threat In Eduroam Academic Networks

In this article, we are going to look at a perilous attack over wireless networks and its potential threats. For those who have an undergraduate degree would be familiar with connecting to eduroam networks - it is indispensable for students as well as professors. Despite its benefits, it such a structure that makes it wide open to unauthorized permissions that might break personal confidentiality. Think of what a person can do if one has your login credentials to access the network. In the future, we might also experience such incidents that a person might graduate by changing their grades.

Before we jump into the subject, I'd like to share a few fundamental information to make you understand the subject and the threats better.

What is Eduroam?

Eduroam is the abbreviation of Education Roaming that uses 802.1x security standards over a RADIUS-based infrastructure. Eduroam aims to ensure that the users of member institutions use the network in any other educational institutions without experiencing any problems. The users of eduroam member institutions can connect to the network in another institution (Guest Institution) which is a member of eduroam with the username and password pair they use to connect to the network in their own institutions (Home Institution).

While the user is in the guest institution and sends a connection request, the authoritative server of the guest institution relays the user to their own authoritative server at the home institution and checks if they are authorized or not. Since all these queries are made through an encrypted tunnel created between the servers, the username and password pair cannot be seen anywhere other than the home server. In this case, all users have to do is to define the eduroam wireless network in the host institution as if they were getting connected to the network of their own institution.¹

Eduroam has a federation hierarchy. There still exist two federations: Europe Eduroam Confederation and Asia-Pacific (APAN) Eduroam Confederation. Eduroam-member institutions send queries to eduroam federations in their own countries, and the countries' federations send queries to the confederations that they are linked to.

Authorization Types

As a result of the general analysis we made about member institutions, we observed that as an authorization type, EAP-TTLS and Inner Authentication are used. At this point, we are going to make a detailed explanation of the general features of protocols. In order to understand the general structure, only summary explanations will be sufficient.

1 <http://www.eduroam.org.tr/whatis.php>

- EAP-TLS: An authorization method that allows only certificate-based authentication. TLS protocol is used for security authentication.
- PEAP: Method that enables the operation of EAP authorization type in a TLS tunnel. Can be named as more of an encapsulation rather than a method.
- EAP-TTLS: Used to provide an EAP on the TLS tunnel.

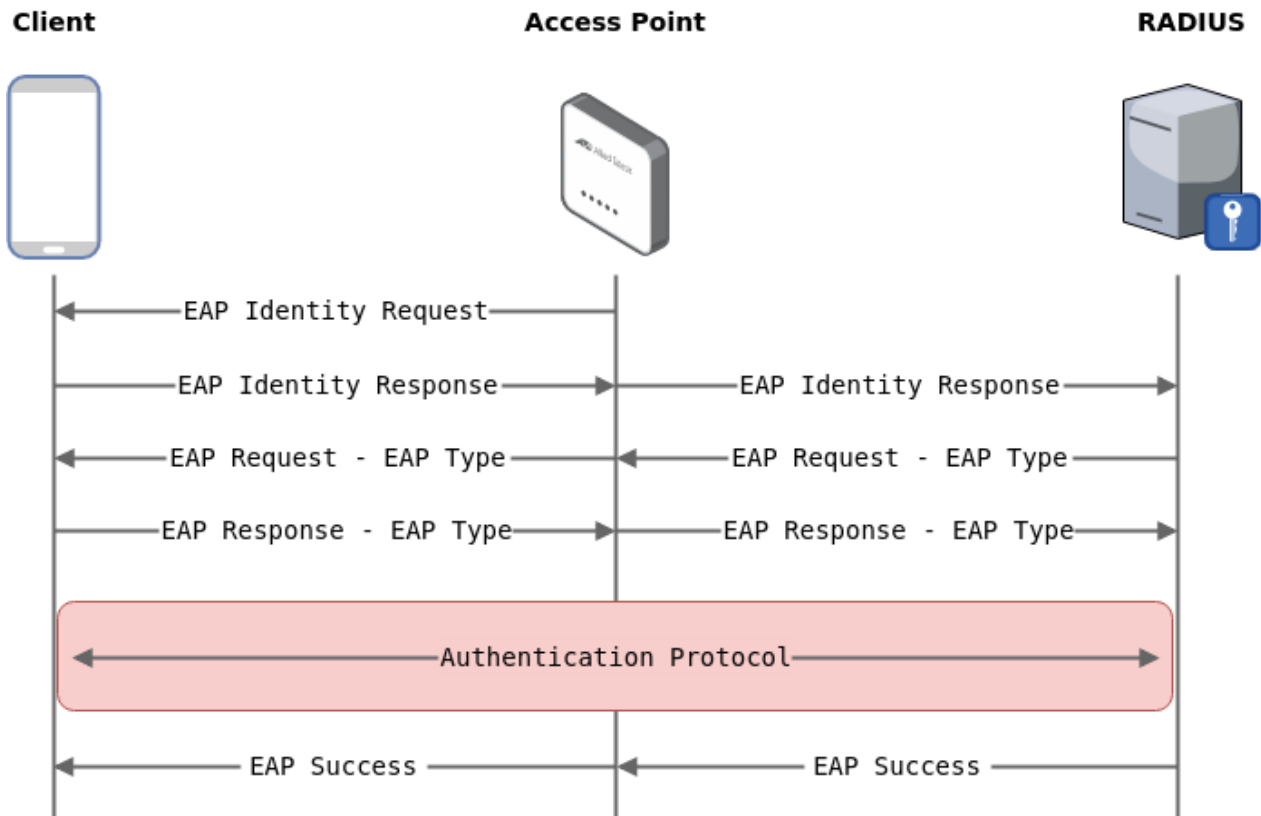


Figure 1 : https://pwn.no0.be/exploitation/wifi/wpa_enterprise/

Inner Authentication Methods

These methods can as well be named as Tunneled Authentication. That is to say, firstly a certificate-based authentication is made, then these methods are used to send username and password.

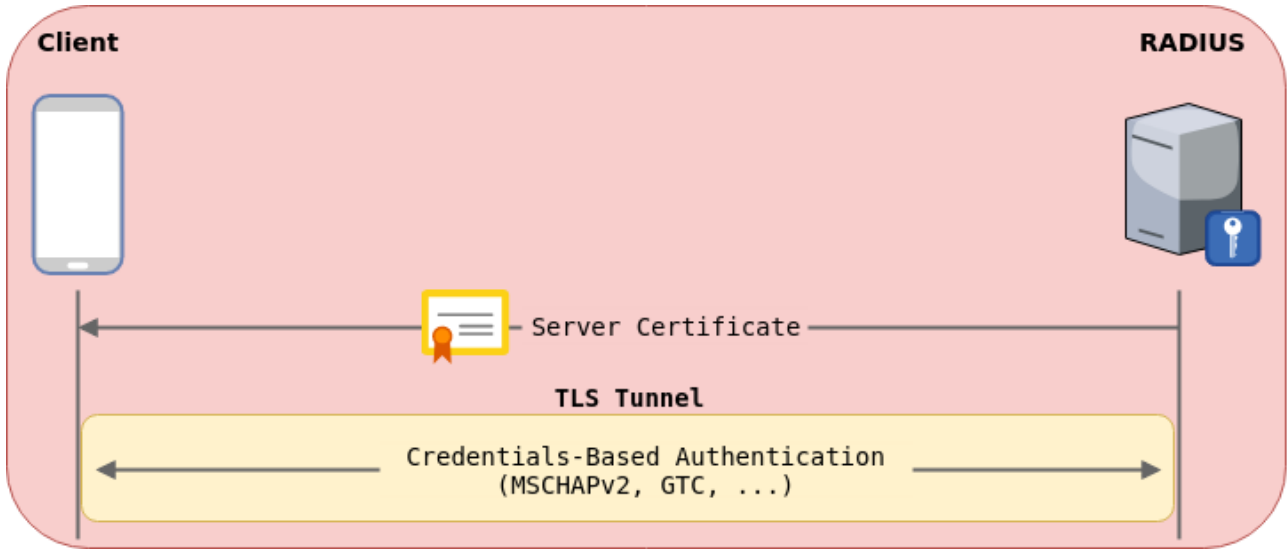


Figure 2: https://pwn.no0.be/exploitation/wifi/wpa_enterprise/

- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)
- Microsoft CHAP (MS-CHAP)
- Microsoft CHAP version 2 (MS-CHAP-V2)
- EAP-MD5 Challenge (EAP-MD5)
- EAP-Generic Token Card (EAP-GTC)

So, what is the problem?

In all these structures, the most problematic point for eduroam users is that as EAP-TTLS and Inner Authentication, PAP is used. What is worse is that there are universities that have put this up in their directives. Below are some exemplary screenshots.

The image shows a mobile device screen with eduroam connection settings. The settings are as follows:

- eduroam
- EAP yöntemi: TTLS
- Aşama 2 için kimlik doğrulama: PAP
- CA sertifikası: (belirtilmemiş)
- Kimlik: eposta@adresiniz@sdu.edu.tr
- Anonim kimlik: eposta@adresiniz@sdu.edu.tr
- Şifre:
- Şifreyi göster

Below the settings is a screenshot of the ULAKBİM eduroam AYARLARI (Settings) page. The text on the page is as follows:

ULAKBİM eduroam AYARLARI

ULAKBİM bünyesinde yetkilendirme yöntemi olarak EAP - TTLS - PAP kullanılmaktadır. Ağa bağlanacak cihazın kablosuz ağ ayarlarında EAP yöntemi olarak TTLS ikinci seviye yetkilendirmede PAP seçilmelidir. Kullanıcı adı sonunda muhakkak kurum alan adı girilmelidir (ULAKBİM kullanıcıları için @ulakbim.gov.tr). Anonim kimlik bilgisi, ziyaret edilen kurum loglarında kullanıcı adınızın yer almaması için önem taşımakta olup, isteğe bağlı bir ayardır.

Microsoft Windows 8 ve 10 işletim sistemlerinde EAP-TTLS-PAP ayarları otomatik olarak tanınmaktadır. Ancak Microsoft Windows 7 ve öncesi işletim sistemlerini kullanan kullanıcıların, IEEE 802.1x ve EAP-TTLS ayarlarını girebilmeleri için SecureW2 yazılımını kullanmaları gerekmektedir. SecureW2 istemcisinin aktif olabilmesi için, kablosuz ağdaştrıcı yöneticisi olarak Windows seçilmeli, 3. parti yazılımlar devre dışı bırakılmalıdır.

Aşağıda, farklı işletim sistemleri için ULAKBİM ağına nasıl bağlanılacağını anlatan açıklamalar, örnek teşkil etmesi bakımından verilmiştir.

- [Microsoft Windows 10](#)
- [Android İşletim Sistemine sahip cihazlar](#)
- [Microsoft VISTA](#)
- [Microsoft XP](#)
- [Linux Genel](#)
- [Intel Proset Kablosuz Programı](#)

Figure 3: Sample connection directives from university and ULAKBİM institution web sites

```

<false/>
<key>EncryptionType</key>
<string>WPA</string>
<key>EAPClientConfiguration</key>
<dict>
  <key>TLSAllowTrustExceptions</key>
  <true/>
  <key>TTLSInnerAuthentication</key>
  <string>PAP</string>
  <key>EAPFASTUsePAC</key>
  <false/>
  <key>EAPFASTProvisionPAC</key>
  <false/>
  <key>EAPFASTProvisionPACAnonymously</key>
  <false/>
  <key>AcceptEAPTypes</key>
  <array>
    <integer>21</integer>
  </array>

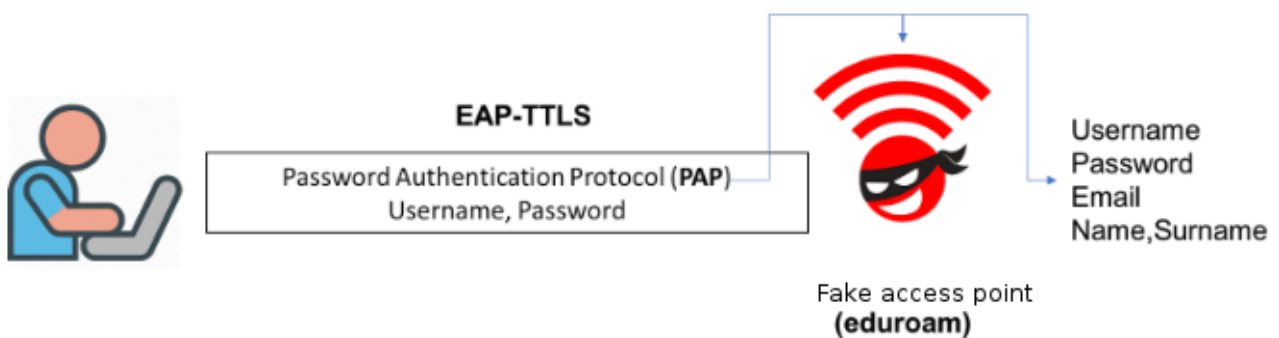
```

Figure 4: Configuration files of Apple devices

When PAP is examined further, it is seen that PAP transmits the information openly from the RFC1334 standards.

However, it is not possible to directly catch these data. As explained earlier, a safe tunnel is created for this data with TLS tunneling method. Actually, this weakness on the PAP side has sort of been remedied this way. If you tried to listen to the surroundings by turning the monitor mode on of a network card, you wouldn't be able to see clear-text data.

Yet, there is a way to display these data openly, which involves being at the end of the communication. Point to Point. An attacker who located oneself at one end of the communication can obtain almost everyone's username, email address and password info who were previously connected to the eduroam network.



Sit at a cafe crowded with college students and open a fake Enterprise network and name it eduroam - the results are terrible because you'd be able to see such information as email, student number, username, and password effortlessly.

Here's a scary example:

```

wlan0: CTRL-EVENT-EAP-PROPOSED-METHOD vendor=0 method=25
wlan0: STA [redacted] IEEE 802.1X: Identity received from STA: '[redacted]@chacettepe.edu.tr'
wlan0: STA da:25:08:0b:7d:dd IEEE 802.11: disconnected

```

Extraction of username, email, university, and password information:

```
eap-ttls/pap: [REDACTED]
username: [REDACTED]@metu.edu.tr
password: [REDACTED]
-1--0 CTRL EVENT END FAILURE 10 75 44 70 35 -1
```

Possible Dangers:

When listing the possible dangers we actually can actually talk about infinite numbers of attacks. However, let's count a few that ring our bells the most:

1. Privacy diminishes and someone who does this can access others' accounts.
2. Those who access to others' accounts may be able to change grades.
3. There is a risk of infecting everyone with malware if the password of a lecturer or the rector is obtained.
4. Chaos may occur between lecturers.
5. If the exploited password is used elsewhere, thine other accounts might also be affected.

As a solution, since the part that causes the problem here is PAP: using PAP for Inner Authentication should be given up.

The Ship That Got Hacked Ashore

Sea transports cover almost 80-90 percent of the world trade transport. Biggest elements of these maritime lines are apparently ships. So can these ships be hacked in any way and what happens if they were to be hacked? Can hacking merchant ships and passenger ships cause serious loss of life and property damage? In this article, we are going to try and find answers to these questions.

Ship hacking had been getting quite popular in the last few years. There are conferences, seminars, and researches held all over the globe.



Sea trades having a high trade volume and an increase in the number of accidents in the last few years aroused my curiosity and made me wonder if these incidents could have occurred as a result of a manipulation caused by hacking. The researches I made afterward showed me that if ships or related peripheral systems are hacked, such incidents as explosions, collisions, sinking because of load balance and route deviations may occur. So while a person waits for their lover's ship, they might encounter with a completely different ship that lost its direction.

So what elements on the ship or around the ship's environment can cause this hacking? Let's list and examine them:

- Human Factors
- AIS Transponder
- ECDIS
- GPS
- Serial Networks
- EDIFACT Messages
- Sea Satellites
- VoIP Radios
- Autopilot
- BNWAS

We'll take a look at these elements and the risk factors they create. Before starting to discuss technical tools and issues, I'd first like to talk about how human factors can be manipulated on ships just as they do in every system.

Human Factor and Sea Satellites

There are numerous sources available on my blog as well as on the internet about reconnaissance about humans. Yet, aside from elements of general information collection, we are going to mention how the information about the ship can be used. Satellites for the seas are produced by many companies and organizations. Examples of the most well-known and preferred are Cobham, KVH, Inmarsat Solutions, Telenor Satellite. Let's do some searching with our fellow Shodan.

When we searched Shodan for Cobham's (one of the companies mentioned above) Sailor 900 receiver, we came up with the following result:

The screenshot shows a search on Shodan for "sailor 900". The interface includes a search bar with the query "title:'sailor 900'", a navigation menu with options like "Exploits", "Maps", "Share Search", "Download Results", and "Create Report", and a sidebar with "Show API Key" and "Help Center".

TOTAL RESULTS: 40

TOP COUNTRIES:

United States	18
Australia	6
Canada	5
Norway	3
France	3

TOP SERVICES:

HTTP	23
HTTPS	11
HTTP (8080)	6

TOP ORGANIZATIONS:

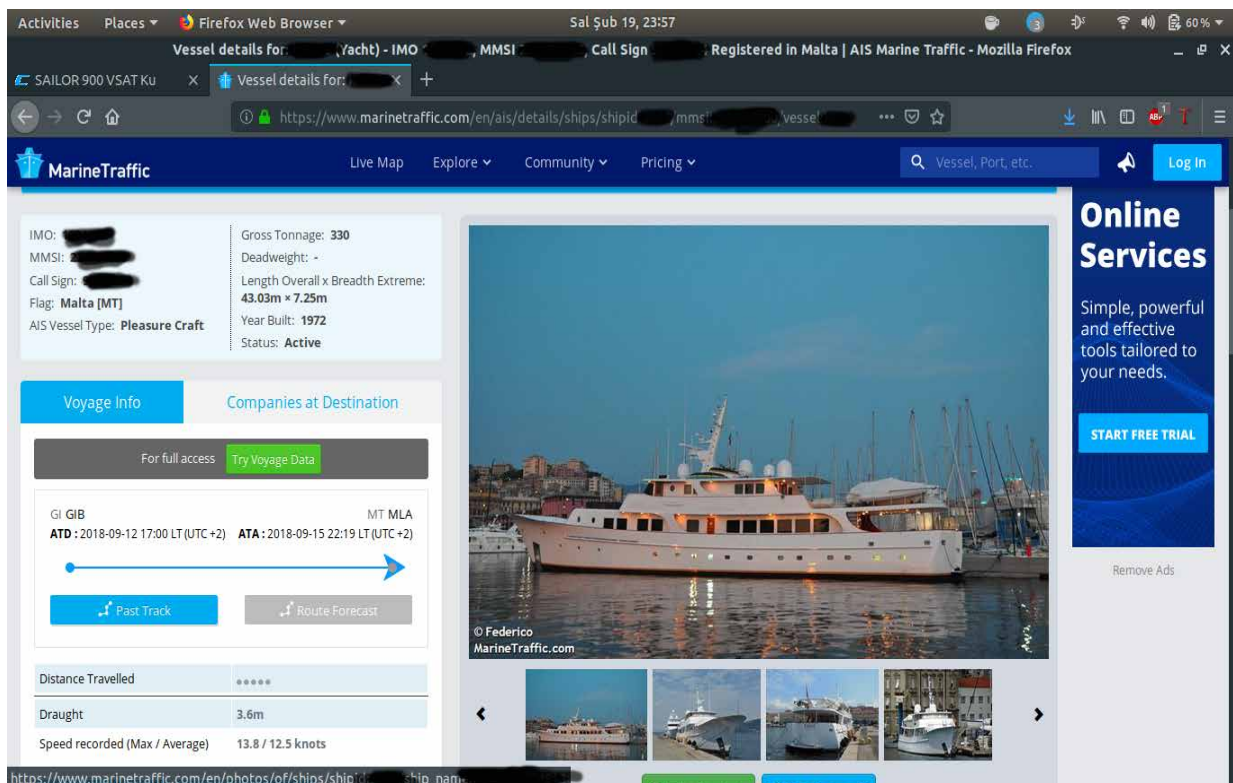
IsoTropic Networks	20
Applied Satellite Technology A...	6
Telenor Satellite AS	3
Intelsat Global Services Corpo...	3

The search results list several "SAILOR 900 VSAT Ku" entries. Each entry includes an IP address, a company name, and HTTP headers. For example, the first result is from Geolink Satellite Services SAS (France) with IP 185.7.14.19, added on 2019-02-18 19:09:25 GMT. The headers include: HTTP/1.1 200 OK, Expires: Mon, 18 Feb 2019 19:09:26 GMT, Cache-Control: max-age=0, Content-type: text/html, Set-Cookie: tt_adm=deleted; expires=1, Transfer-Encoding: chunked, Date: Mon, 18 Feb 2019 19:09:27 GMT, Server: lighttpd/1.4.33.

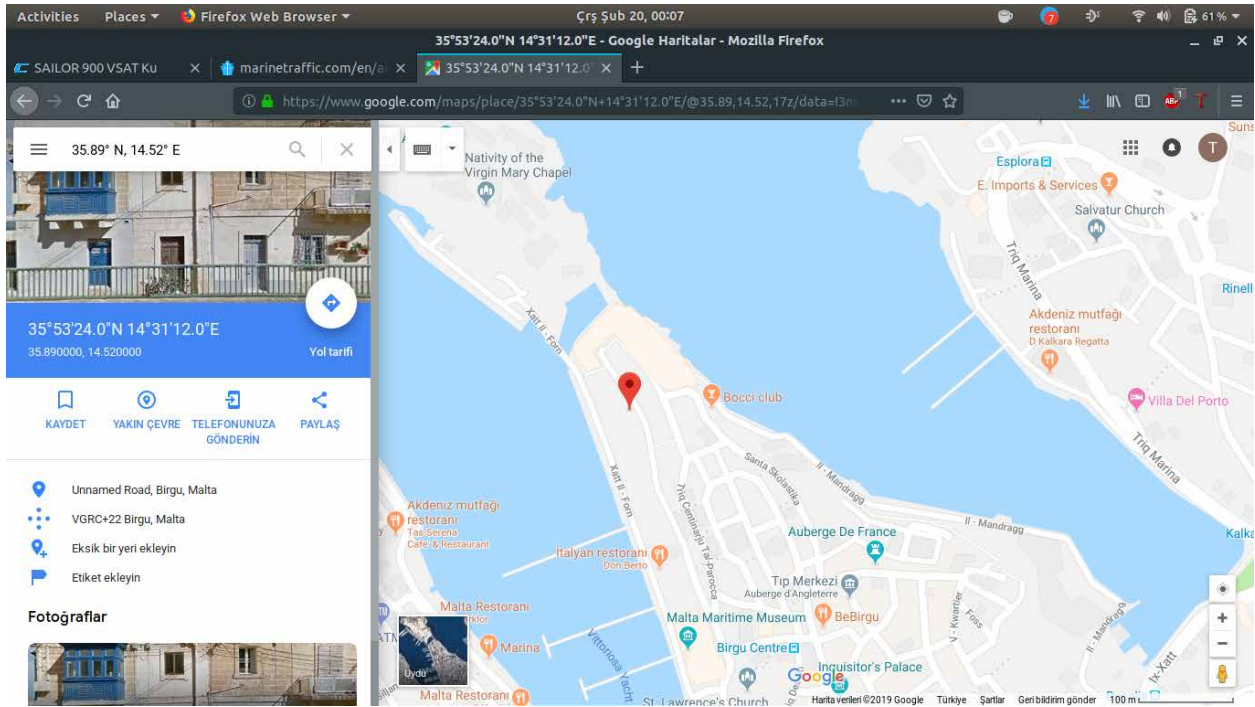
Let's see what the very first server we see can offer us:



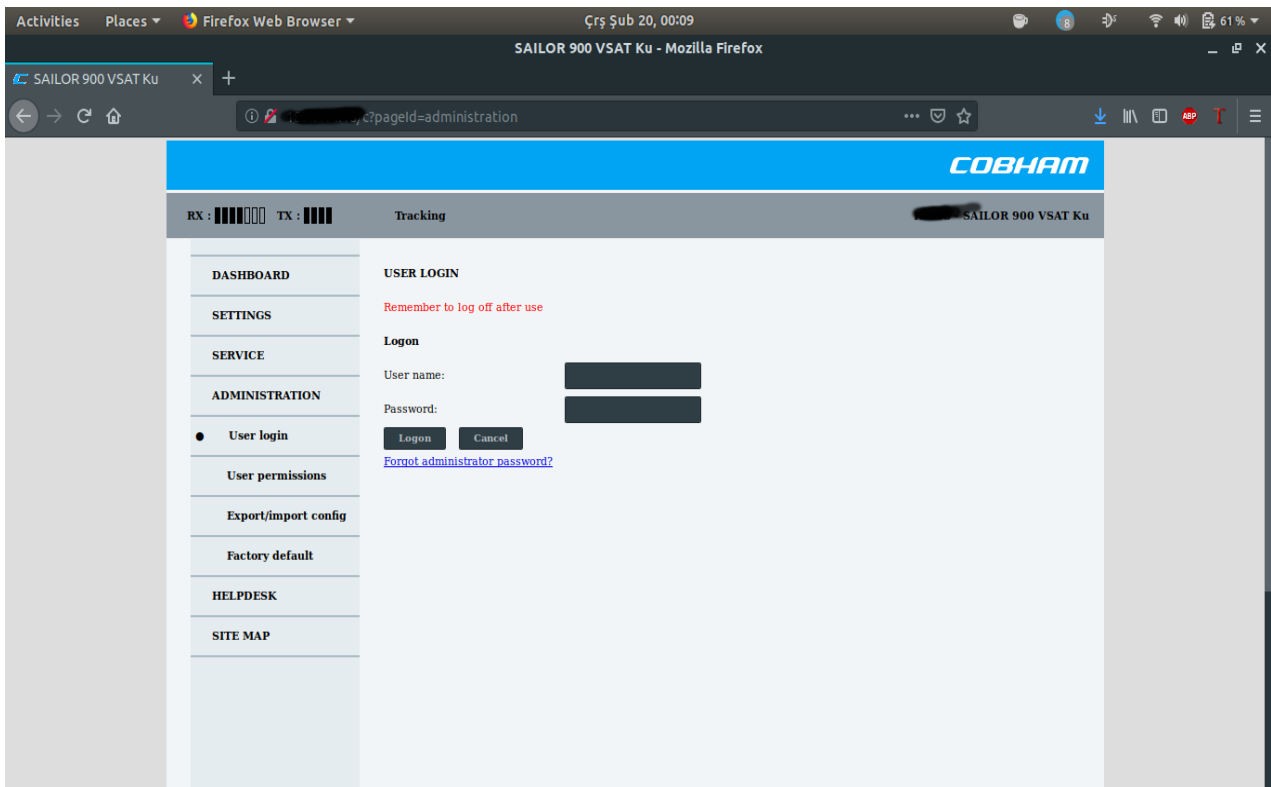
We are greeted with such information as the ship's name, location, and course. Let's search this ship through MarineTraffic.



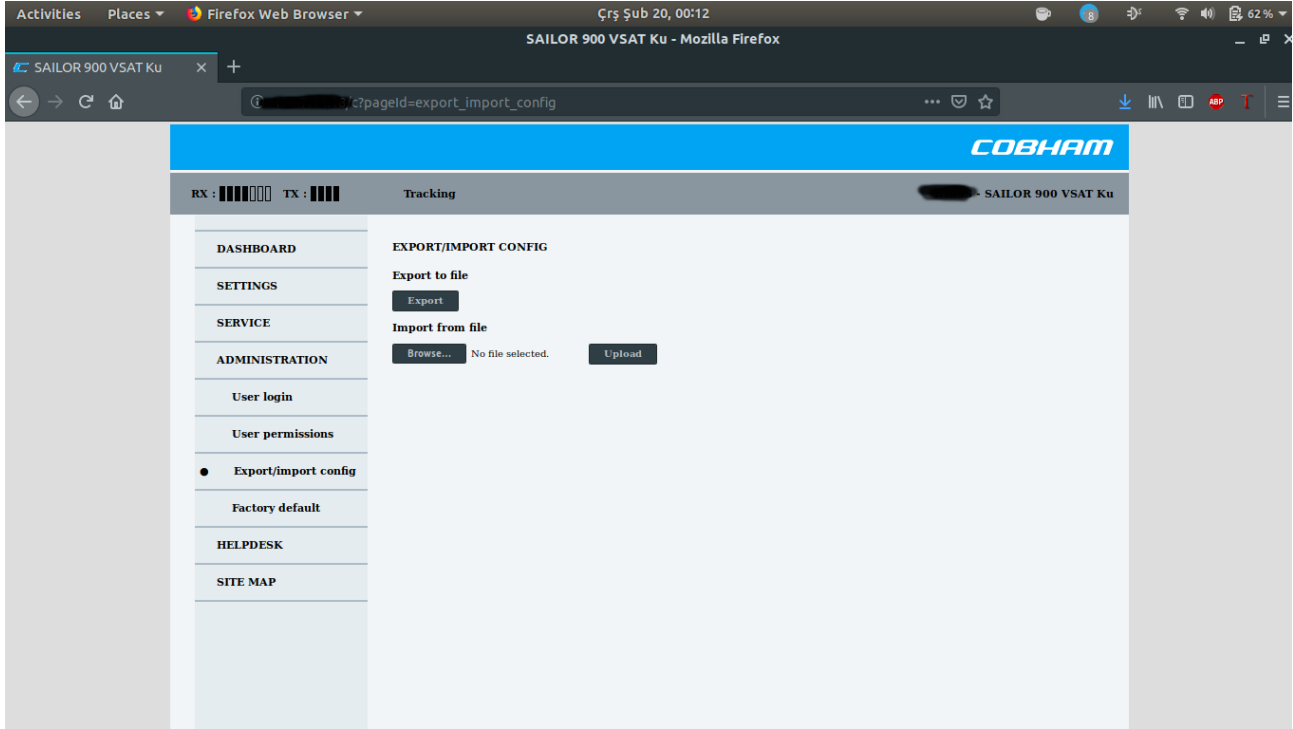
It is a pretty cute yacht. In addition to the images, we were offered the information on the yacht's route, dimensions, call sign and location. Again, in addition to all these, there is something rather more useful for us. The historical change process of the yacht's name and contact information are also available on MarineTraffic. If you wish, you can check if the location information matches the satellite interface we use.



Heading back to the satellite interface. Let's see what we can do with *Administration*.



On the panel, we encounter I tried admin/admin and admin/pass but those failed. Later, I tried the admin/1234 combination and it worked!

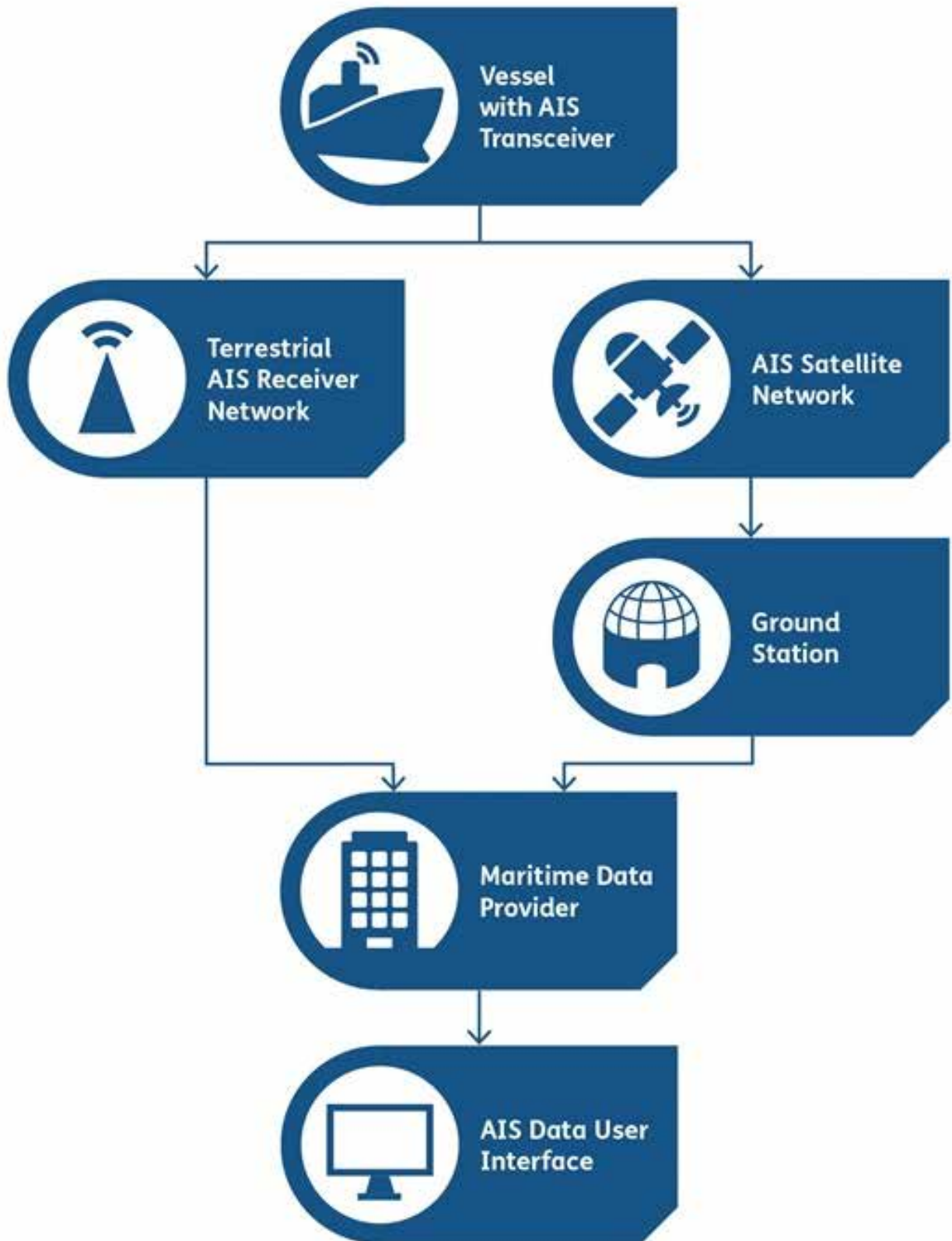


Leaving this screenshot here, you can imagine the rest. What I'd like to point out here is that these satellites have a web interface and that the values used to authenticate in this website are left as default. During reconnaissance, if you tamper with Shodan and make usage of Instagram's location-based search, you can access pretty interesting stuff. There even are selfies who contain even the device information but I'm not going to share them here and leave it once again to your imagination. Let's keep this short and examine other elements.

AIS Transponder (Automatic Identification System)

What is AIS?

AIS is a system that enables the tracking of surface vehicles. Thanks to the system, surrounding ships' route, speed, location, and name can be found. When transmitting, AIS uses VHF Sea Radio frequencies. In order for the broadcast to be watched and decrypted, an AIS Receiver and AIS Transponder are needed. There are two types of AIS transponders: Type A and Type B. Type A transponders are used in merchant ships whereas Type B transponders are used in smaller-scale load ships and cruise liners. While Type A transponders have a range of 15-20 nautical miles, Type Bs have 5-10 nautical miles. Ranges, of course, may depend on the antennae used or whether they are multi-directional or unidirectional.



AIS Manipulation

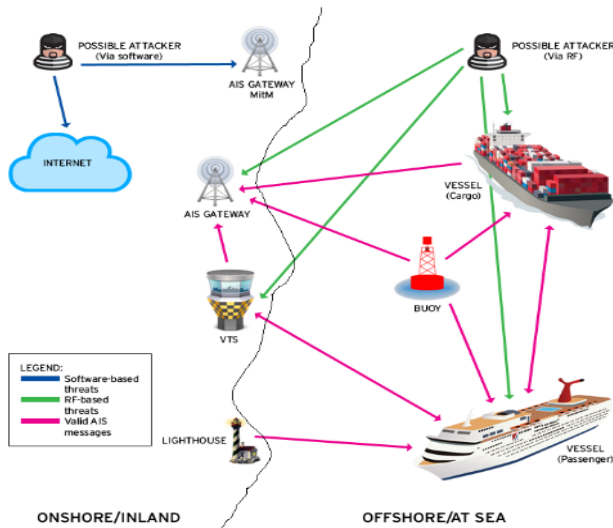


Figure 1: Possible AIS attack scenarios

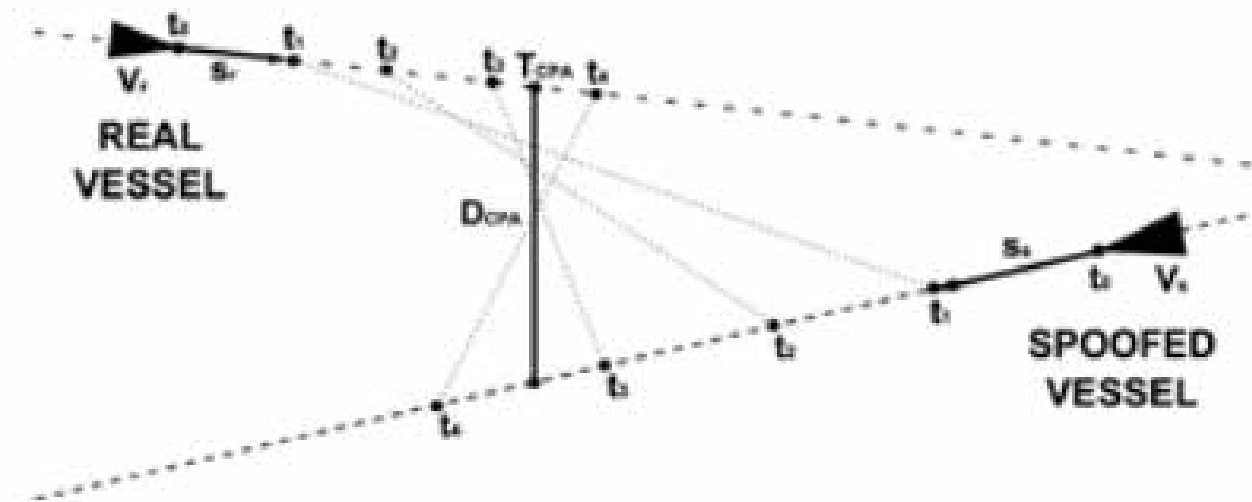
Trend Micro, one of the important companies, published a report named “A Security Evaluation of AIS” in 2014 about AIS security. In this report, facts that may cause AIS manipulation and how these can actually happen have been served in detail. In addition to this, an exemplary manipulation has been made. According to this report there exist two elements which may cause AIS manipulation. One of these is RF i.e., radio frequencies and the other is the IAS software. Possible threats are classified under 3 categories and divided into 3 sub-categories.

Let's examine them:

A) RF-based Threats:

1) CPA Spoofing:

Avoiding collision is one of the basic aims of AIS and when a collision occurs or is expected, a response is automatically made. CPA works by calculating the minimum distance between two moving ships and in any case of collision, can warn the captain visually or vocally.



The figure above shows the working algorithm of CPA. On the figure, TCPA represents the time before arriving at the CPA point, and DCPA represents the distance between the two ships before the arrival to the CPA, and $w(t_i)$ is the

distance left between the two ships at any time t . S_r and S_s are the vectors of the ships. An alarm is given when one of the D and T parameters go below a certain level, that is to say, the level based during the configuration of CPA. CPA spoofing involves making a ship look like a possible collision. This case triggers the alarm and this may cause the ship to hit a rock on shallow water or run ashore.

2) AIS-SART Spoofing

SART stands for Search and Rescue Transponder. It is an active radar reflector who aims for it to be located in emergency cases by sending signals to surrounding ships. The device can be found in emergency boats or persons.



SART spoofing works by creating false distress signals at specified coordinates. The aim is to bring the target ship to the desired coordination - by law, it is mandatory for the ships to attend to the rescue mission if and when they receive a SAR message.

2) False Weather Forecasts

In addition to location information, AIS also gives information about environmental circumstances currents and weather forecast. The purpose of giving false forecasts may be something like displaying a stormy day as a sunny one.

3) Slot Starvation

To give an example, you can think of this attack as the DHCP Starvation attacks. The aim of the attack is to prevent

the stations within ranges of each other from communicating. These stations include vessels used in traffic monitoring, devices that allow *AtoNs* (Aids to Navigation - i.e., navigation helpers like lighthouses) to be specified in the AIS and AIS network gateways. As a result, attackers can disable AIS to a large extent.

4) Frequency Hopping

By mimicking the nautical authorities and instructing one or more AIS transponders to change the frequency they work on, AIS can be dismissed with different frequency broadcast. It is by law that the receiving stations must execute every single instruction given by the nautical authorities. This is why these attacks are still made. Restarting the system doesn't help much since the frequencies are changed only when an instruction is given.

5) Timing Attacks

By overriding the commands, attackers can tell the transponders to delay the transmission times thus blocking the data transmission about where the ships are. This case causes the ships to disappear from the AIS radars. Besides, contrary to slowing down the transmission time, attackers can cause overloading of the maritime traffic by allowing continuous location information transmission and frequent status updates.

B) Software and RF-based Attacks:

1) Ship Spoofing

This case represents showing a non-existent ship as if it did. Lots of fake information are created such as ship name, speed, destination, route, and flag.

2) AtoN Spoofing

AtoNs exist to aid the vessel traffic management throughout channels or ports. Or else to warn the captain against the dangers in offshore waters, shallow waters, or rocky outcrops. AtoN fraud is based on the process of producing fake data to manoeuvre ships. Various scenarios can be created by using stimulants like fake buoys and also adding Ship Spoofing.

3) AIS Hijacking

AIS Hijacking involves changing any information about current AIS stations. Attackers can alter data about AtoNs. On the software side of Hijacking, attackers can listen to ongoing communication (MITM attacks) and alter AIS information. On the RF side, they can override AIS messages by sending higher frequency messages.

C) Software Based Attacks

Before talking about the attacks, let's give brief information about AIVDM. AIVDM is the protocol AIS uses in the application layer. AIVDM has 27 types of messages, each with its own specific purposes. AIVDM has 27 types of messages, each with its own specific purposes.

Category	Message	Description
Standard	1	Scheduled position report (class A)
	2	Assigned position report (class A)
	3	Special position report (class A)
	5	Static report (class A)
	9	SAR aircraft position report
	18	Position report (class B)
	19	Extended position report (class B)
	24	Static report (classB)
	27	Long range position report
AtoN	21	AtoN report
Timing	4	Base station report
	10	UTC inquiry
	11	UTC response
Safety	12	Addressed text message
	13	Acknowledgment
	14	Broadcast text message
Binary	6	Addressed binary
	7	Binary acknowledgment
	8	Broadcast binary
	17	GNSS update
	25	Short binary (no acknowledgement)
	26	Binary with communications state
Other	15	Interrogation for specific messages
	16	Assignment mode command
	20	Data link management
	22	Channel management
	23	Group assignment command

AIVDM is a two-layer protocol. The outer layer is a variant of NMEA 0183, an old standard for data exchange between navigation systems.

Trend Micro, which examines software-based attacks, targeted MarineTraffic.com, AIS Hub and VesselFinder, 3 of the greatest AIS providers in their report. In their analysis, they concluded that the providers did not check their sources and did not examine whether incoming messages actually came from the sender ships. In addition, researchers who have stated that there is no way to verify the sender identity in AIVDM and that this causes attacks like MITM, have even made an exemplary scenario themselves.

Firstly, false AIVDM messages were created using AIVDM encoder and these messages were of type 21 for AtoN reports and of type 13 for buoys. Afterwards, they created a false report for an anchored ship and sent it to the AIS provider via email.

```
To: report@marinetraffic.com
```

```
MMSI=247320161
```

```
LAT=44.3522
```

```
LON=8.5665
```

```
SPEED=0
```

```
COURSE=243
```

```
TIMESTAMP=2013-11-11 13:11
```

Finally, they created a script to send false data to the AIS station, creating a data stream showing the so-called presence of a vessel on the route of the word PWNED in the Mediterranean Sea.



Now that we have done examining AIS, let's take a look at ECDIS in which AIS is used integrated with.

ECDIS (Electronic Chart Display and Information System)

What is ECDIS?

ECDIS is a sea map system alternatively to paper maps. It can be used as a standalone map display function or integrated with sensors of additional navigation systems such as GPS, AIS, and radar, which is how it is often used. ECDIS uses two types of data when performing this operation. These are ENC (ESH) and RNC (RSH). These are the data types In this article, we will focus more on these data types. Let's examine further of what ENC and RNC are.



RNC: RNCs are the digitalized versions of scanned paper maps. Can be used by the ECDIS in cases where ENC's are not accessible.

ENC: ENC's are digitally prepared vector drawings and make up the main data source of the ECDIS. Prepared and made available according to IHO's (International Hydrographic Organization) standards. RENC's step in about the distribution of ENC's. Nowadays, there are two RENC's and these institutions work integrated with each other. One of them is the IC-ENC in the UK and the other is Primar-Stavanger in Norway. These institutions take the ENC's from the ENC manufacturers linked to them and send them to the distributors linked to them again. Here, by manufacturers I mean countries. For instance, there are 43 countries found in the IC-ENC. When the distributors are examined, we encounter with 7 institutions as seen below.



Distribution of ENC's

The main risk factor in this system is the distribution of ENC's. Today there are ships who receive ENC distribution via internet connection. For instance, one of the distributors, Primar, provides ENC to ECDIS online. So, what kind of security problems might this online distribution cause?

PRIMAR » Services » ENC Distributors » Online distribution

BENEFITS

- Reduction in effort and cost
- Choose your service level
- Direct permit access
- Increased customer satisfaction
- Available around the clock
- No shipment delays

ONLINE DISTRIBUTION SOLUTIONS

DESCRIPTION

Our online services can be used instead of or as a supplement to our ENC CD service. They are intended to facilitate automated processes and provide flexible and efficient methods for ENC distribution.

Our available online services are as follows:

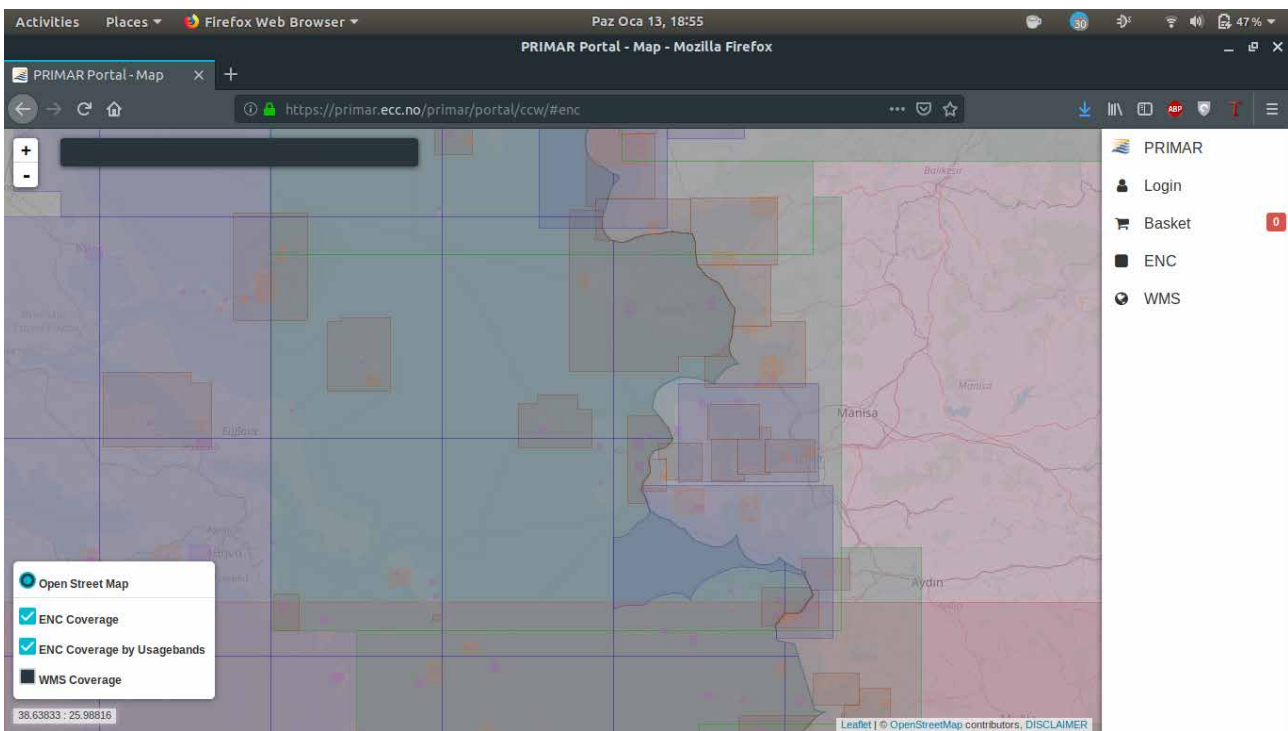
PRIMAR online using the Chart Catalogue
PRIMAR online is an integrated part of the PRIMAR Chart Catalogue for downloading of ENC's and permits. Supported media like memory stick or CD is used to transfer the ENC's into the ECDIS/ECS for updating the portfolio of ENC's.

PRIMAR online using e-mail
PRIMAR online e-mail notification is an independent web service for downloading of ENC's and permits. The customer will regularly receive an e-mail including a link to a web page where licensed ENC's and permits are available for download. Supported media like memory stick or CD is used to transfer the ENC's into the ECDIS/ECS for updating the portfolio of ENC's.

PRIMAR online using ECDIS
ECDIS online is an internet-based service for maintaining a vessel's ENC portfolio. In this service the customer has functionality in its ECDIS/ECS to directly interface and download ENC's and permits from PRIMAR. Distributors or OEMs can contact PRIMAR to receive copies of relevant interface protocols. The protocols support deliveries using http and e-mail communication.

Even though ENC distributions are made via CD/DVD or USB, another means of distribution is through a web application and, for example, as Primar stated on their website, HTTP and email are used to data transfer. I seem to see the eyes of the readers who see HTTP shining while reading the sentence. In case of an ENC supply or an update, if included into the ship's network and performed an MITM attack, instead of using Primar's service as a provider, it would be a major problem for security if an ENC provided and compiled by a service created by the attackers is included into the ECDIS system. On the other hand, connecting to the internet and installing a misconfigured ENC or software update would also be a major risk factor. You can visit the link below as an example to the web interface.

<https://primar.ecc.no/primar/portal/ccw>



To ensure security about this subject, ENCs are encrypted according to the S-63 standard specified by IHO to prevent them from being copied or distributed illegally. According to the standard, the database used in the data flow is encrypted using the Blowfish algorithm, hashed using SHA1 and additionally CRC32 control system is included. Based on the same standard, signatures are defined in DSA format for the users to decrypt and use the data. However, the standard is unfortunately poorly implemented by ECDIS manufacturers.

Risk factors for ECDIS

- CD/DVD or USB devices used in ENC distribution may contain malware, in which human factor is to blame the most. This way, even though the malware is not directly installed, it can be connected to the network as an intermediary, whereby a download and run operation or a faulty software update installation can occur.
- Theft or modification of ENC in online ENC distributions.
- Receiving data from sensors and misrepresenting them to ECDIS.
- Access to other devices in the LAN, Wi-Fi access points or other computers may be ensured by connecting to the internet over ECDIS.

Now it's time for another element which can be used integrated with ECDIS: GPS. Let's talk a bit about them.

GPS Spoofing

GPS spoofing has been among the most popular subjects in the last few years. The fact that two patrol boats belonging to the American army entered the Iranian territorial waters and deviating from the routes of nearly twenty ships in the Black Sea in 2017 and many other similar cases made GPS spoofing a worldwide agenda.

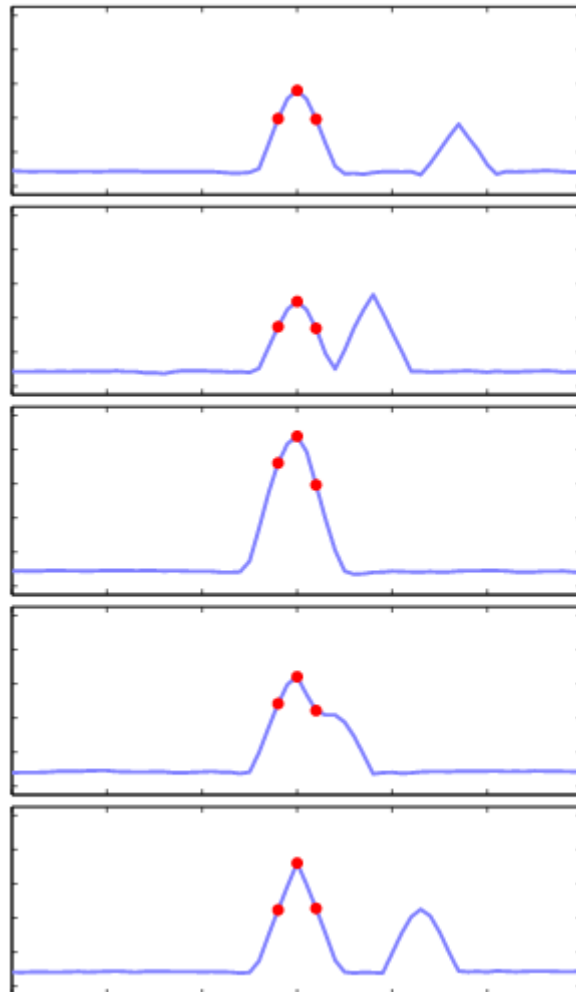
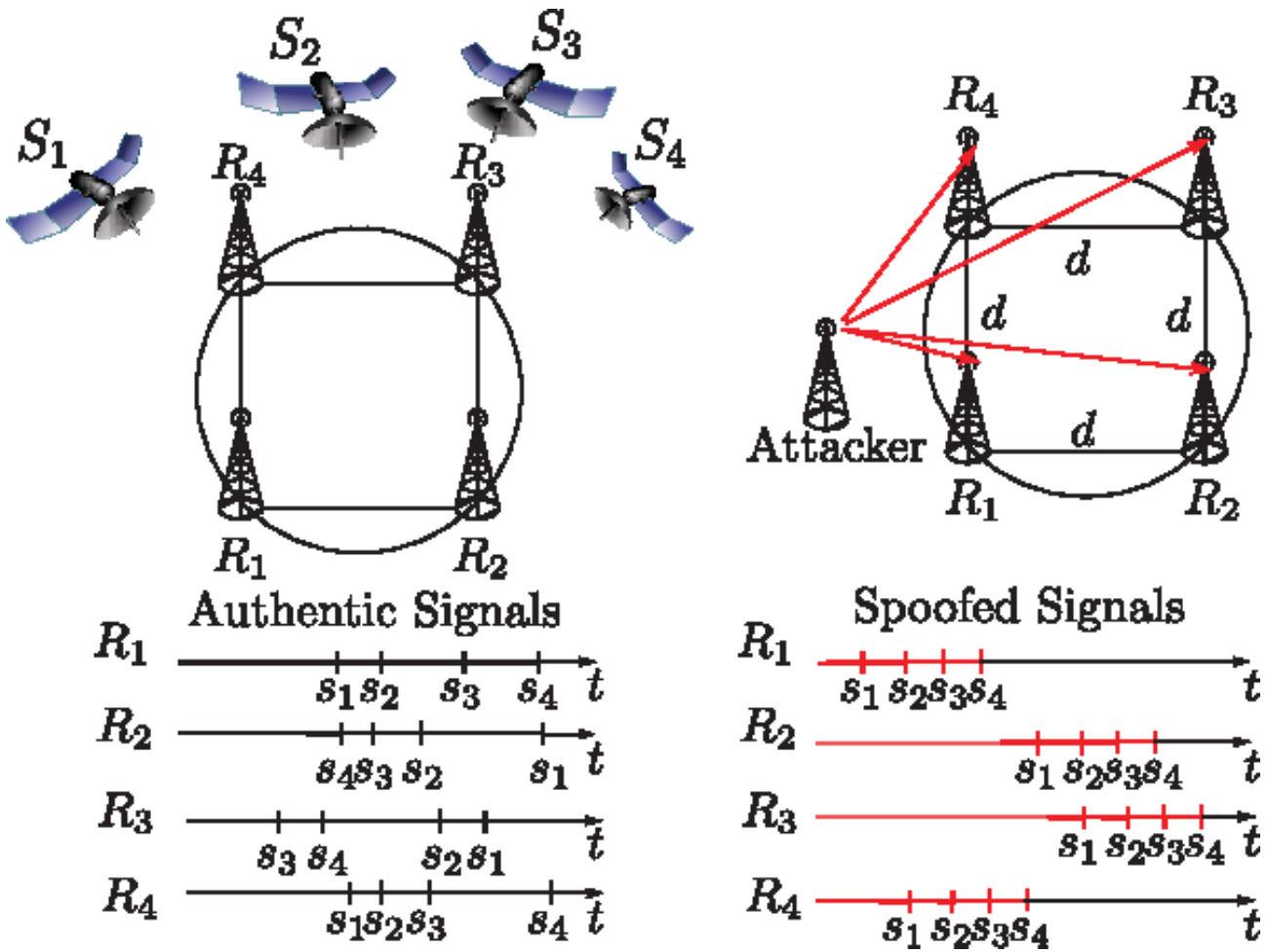


Figure: A successful GPS Spoofing Attack made against one channel

What lies on the base of GPS spoofing attacks are sending false GPS signals to the target as if they were real GPS signals or sending prerecorded real GPS signals later. Generally, signals very similar to what the target usually receives are sent as a beginning and as moved further, the power of the false signals are increased step by step and the device stays connected to these signals.

Hardware such as USRP B210 is used in attacks involving the recording of a pre-existing signal, whereas in the case of repeating the signal, equipment such as bladeRF is used.



If you want to create the signal yourself instead of recording it beforehand, you'll need to specify the parameters and path you are going to use for simulation. After, generate an IQ data file using software like GNSS Signal Architect. Then, send the generated IQ data file to a device like the USRP N210 for radio frequency transmission. Of course, this is not that easy, there are lots of technical details. However, late use of a recorded real signal is a bit easier and is more common.

Speaking of GPS spoofing, we need to talk about the transfer of GPS data from the GPS antenna to the computer screen.

Serial Networks

Ships typically have 2 kinds of networks. The first is the IP/Ethernet network used for sectoral systems, crew mails, and web services. The second is the Serial Network which is used for operational technologies (OT), including rudder, engine propulsion, stability, and navigation.

What is a Serial Network?

A serial network is a network in which serial communication is used, a form of communication in which the data is transmitted in a serial manner, ie the bits in a data are sent over the same line. There are two frequently used standards in this communication type: RS-232 and RS-485.

How are Serial Networks Hacked?

So how do we connect to a serial network on the ship? There are bridge points for switching between IP and serial networks. In order to connect to the serial network, you need to know which devices have a bridge between the serial network and the IP network. Lots of exemplary devices can be found on ships. As an example, we can give ECID, AIS Transponder, and serial-IP converters.



Exploiting the Converters

Moxa, Perle and such other converters are used to send serial data with IP/Ethernet network cables. We can show 3 ways to exploit these converters:

1) Default Converter Passwords

Converters usually have a web interface for configuration. The information for these interfaces are usually the default usernames and passwords the manufacturers publish on their websites.

2) Exploitable Converters

Some Moxa converters have been developed to be exploited. As an example, we can give Metasploit modules developed for CVE-2016-9361 vulnerability. With these modules, attackers can determine or learn themselves by performing an admin password recovery.

3) MITM Attacks

For example, this attack may be performed on the data stream from the GPS. By performing an ARP Poisoning attack on the network, routing can be done over the attackers' computer.

Talking about communication, let's take a look at the communication between the port and the ship.

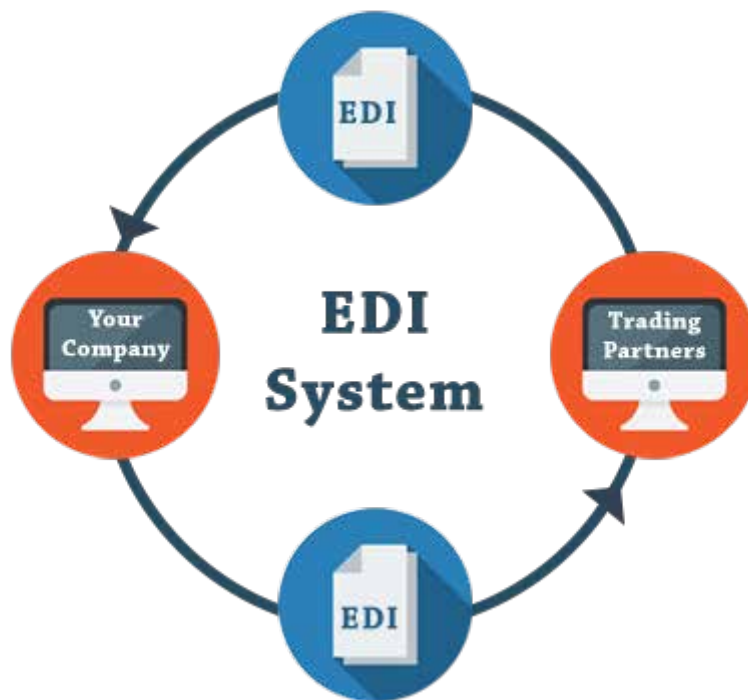
EDIFACT Messages

What is EDI (Electronic Data Interchange)?

EDI is a set of standards established to ensure the speed, security, and control of the transmission of these documents and data, which enables the electronic transmission of commercial documents. This system exists in almost every sector trade is and is even used in non-trade data applications. So far, we have mentioned standards - but what exactly are them?

After the emergence of the concept of EDI, many standards have emerged. The most well-known of these are the North American ANSI X12 and the UN UN / EDIFACT standards. These standards cover the scope and method of data transmission.

How does EDI work?



There are 4 main elements to EDI's working: Standard, Data Conversion, Mapping, and Communication. Let's take a look at these elements.

1) Standard

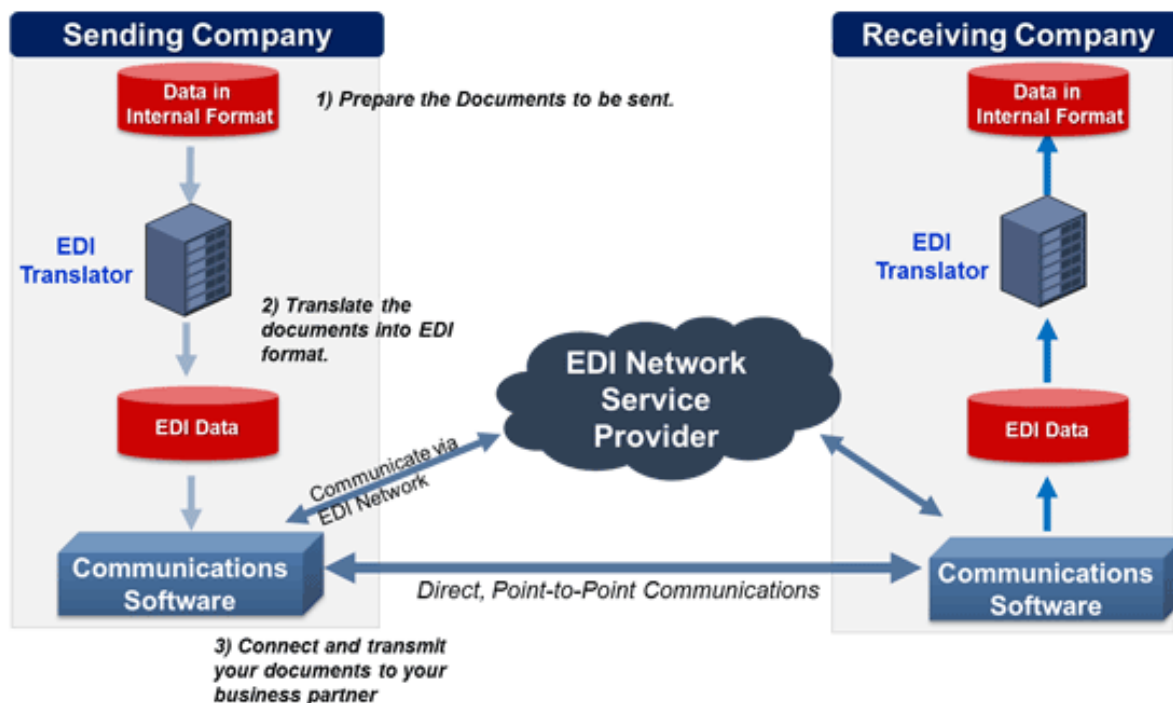
Set of rules created like EDIFACT, X12, AIAG. The standard in which EDI will be used is determined at this stage.

2) Data Conversion

The process in which any processed document is converted to EDI format or a document with EDI format converted to a readable format. The conversion is done using the EDI converter, according to EDI standards.

3) Mapping

Converting the EDI-format data into data types that are easier to use such as .txt, .xml.



4) Communication

This stage is where the transmission type to the target is determined. For this, there are two options: VAN (Value-added network) and direct communication. You can think of VAN as a customized network communication pool. These pools are networks enterprises build for communication. In direct communication, protocols like FTP(S), HTTP(S), SMTP, and AS2 are used.

How do EDIFACT Messages Pose a Risk?

Here, you might have some questions in your mind about the relationship between EDIFACT messages and ship hacking and posing a threat. Let's try to wipe those question marks away. Lots of data are found in these messages: what kind of loads are going to be loaded onto the ship, the load balance of the ship, etc. These data belong to the message group specified as BAPLIE. Manipulations are done to this group.

```

UNB+UNOB:1+PARTNER_ID:ZZ+0038977332:01:MFGB+020331:1230+00000000000001++INVOIC++++1'
UNH+0001+INVOIC:S:93A:UN'
BGM+380+INVOICE-NBR+9'
DTM+137:20000101:102'
RFF+ON:CUST_ORDER_NO'
NAD+RE+::92++MANUFACTURER_NAME'
RFF+VA:DE12931720 6'
CTA+AR+:JANE DOE'
COM+00 49 89 9933-2543:TE'
NAD+ST+::92++COMPAQ COMPUTER CORP.'
NAD+BY+::92++COMPAQ COMPUTER CORP.'
CUX+2:USD:4'
ALC+C++6++ABG'
PCD+1:2.5'
MOA+204:200.00'
LIN+1+10+240152:AB'
QTY+47:3.00:EA'
PRI+AAA:1310.00:CT'
UNS+S'
MOA+77:4378.28:USD'
TAX+7+VAT+++:::15+S'
MOA+176:248.28:USD'
UNT+22+0001'
UNZ+1+00000000000001'
    
```

So, by manipulating these messages, a ship may be sunken, exploded or the loads might get stolen. Here, I'll show an example manipulation of how a possible explosion or fire can be caused on a ship. For this, we're going to benefit from the samples in the code table SMDG published.

For the whole table, see the sources list.

SMDG		v201501 (January 2015)		
SMDG code lists for ATT segment in DGS group (this sheet contains the definition of 4 code lists)				
ATTRIBUTE TYPE (composite C955)				
codes to be used in C955.9021				
SMDG DG Attribute Types List		DE 1131 = DGATT Example: ATT+26+UNX:DGATT:306+0403:CVL:399'		
Code	Name	Description	last change	valid from
AGR	Aggregate State	Aggregate state of a hazardous substance		2013-09-30
BNR	DG booking reference number	DG item's booking reference		2013-09-30
PSN	Proper Shipping Name	Proper shipping name as defined by IMDG Code		2013-09-30
HAZ	Special Hazard	Identification of a special hazard		2013-09-30
QTY	Special Quantity	being applied		2013-09-30
SEG	Segregation Group	Segregation group as defined by IMDG Code		2013-09-30
TNM	Technical Name	Technical name, if different from proper shipping name		2013-09-30
UNX	UN-number extended information	Code as generated by Exis Ltd.		2013-09-30
ATTRIBUTE DETAIL (composite C956)				
codes to be used in C956.9019				

Let's examine the sample message.

ATT+26+AGR:DGATT:306+G:DGAGR:306'



Attribute Type



Attribute State



Substance Type (Gas)

The message seen here states that the cargo to be loaded on the ship is in the gas state.

ATT+26+AGR:DGATT:306+XS:DGAGR:306'

Here, by changing the state of the substance, we have stated that the substance is explosive. If the material to be loaded on a ship is explosive, the code "XS" must be included in the message, but we can change it to define the explosive as a safe liquid. For this reason, the ship will sail before the security measures regarding the transport of explosives are taken. This can be done as follows.

ATT+26+AGR:DGATT:306+L:DGAGR:306'

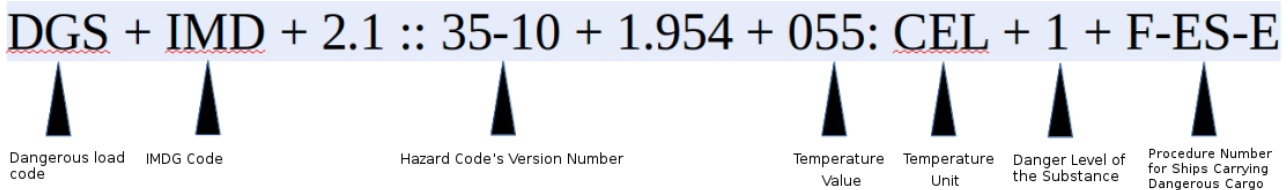
Just as we changed the load type, we can also change the attribute detail. For this, we will play with the code "HAZ" like this:

ATT+26+HAZ:DGATT:306+FLVAP:DGHAZ:306'

Here, we have stated the load as explosive steam by changing the code "FLVAP". If the load to be transported is a combustible material, we can jeopardize the sail by changing the message where the temperature at which the load reacts is indicated.

As you can see in the message, changing the temperature value or replacing CEL code with FAH (representing Celcius and Fahrenheit respectively) would be a major threat for the ships sail.

DGS + IMD + 2.1 :: 35-10 + 1.954 + 055: CEL + 1 + F-ES-E



▲	▲	▲	▲	▲	▲	▲	▲
Dangerous load code	IMDG Code	Hazard Code's Version Number	Temperature Value	Temperature Unit	Danger Level of the Substance	Procedure Number for Ships Carrying Dangerous Cargo	

In addition to explosion and fire events, message manipulations can be performed in relation to the deterioration of the load balance. For more information, see BAPLIE messages.

Elements I'll cover end here. Of course, there will be more to the elements that I could not mention but listed at the beginning of the article. The purpose of writing this article was to raise awareness on this subject and to shed light on its detailed research. I hope the things we skipped will be searched by our curious fellows and results shared with us.

Source:

http://iho.int/iho_pubs/standard/S-63/S-63_e1.1.1_EN_Apr12.pdf

<http://smdg.org/assets/assets/BAPLIE3.0.1e-MIG.pdf>

<http://smdg.org/assets/assets/BAPLIE3-MIG12-Master-Document.pdf>

<http://www.gemitrafik.com/vhf-deniz-telsizi/epirb-ve-sart-nedir/>

http://www.shoddb.gov.tr/shoddb_esas/index.php/tr/urunler/haritalar/elektronik-seyir-haritalari

<http://www.smdg.org/assets/assets/SMDG-CODES-FOR-DGS-ATT-v201501.xlsx>

<https://en.calameo.com/read/004474480397d2632c1e3>

<https://tools.ietf.org/html/rfc4130>

<https://www.boatus.org/study-guide/navigation/aids/>

<https://www.edibasics.com/what-is-edi/>

<https://www.pentestpartners.com/security-blog/hacking-serial-networks-on-ships/>

<https://www.pentestpartners.com/security-blog/making-prawn-espressos-or-hacking-ships-by-deciphering-baplie-edifact-messaging/>

<https://www.primar.org/distributors>

<https://www.trendmicro.de/cloud-content/us/pdfs/security-intelligence/white-papers/wp-a-security-evaluation-of-ais.pdf>

Client-Side Static Analysis in Web Applications

The term *client-side* in the title refers to analyzing the resources of a web application we open in the web browser environment, within the same browser environment.

While saying the files should be analyzed, without a doubt, I mean Javascript files.

Front-end Javascript libraries are technologies that clearly simplify the daily life of both developers and end users. As such, they became widespread in a short time and turned into the technologies we encounter almost everywhere.

It is almost impossible not to hear front-end Javascript libraries such as AngularJS, ReactJS, Vue.js.

To summarize the questions that we will try to find answers in the following sections of the article:

- What does the spread of technology mean to us in terms of security?
- Does it make it easier for us to look for vulnerabilities?
- How and which weaknesses on the sites should we search?

The simplest answer to the question of what is static analysis is the analysis that is done without running the target software. That is the part that concerns us for now.

We will look for weaknesses in the application by using the development tools on our browser, by using the help of similar tools - without making a request other than while viewing the web application in the browser.

What will we be looking for when performing the static analysis?

In this article, we will not mention Recon methods, but we will be looking for URLs and subdomains. Other environments used for internal purposes, private keys, API endpoints, and files/directories will also be among the important information we are looking for.

The codes that indicate potential hazards (eval, etc.) and out-of-date libraries, especially versions that are known to be vulnerable, will be the other details we seek.

Collection of application source files

While browsing the application in a browser environment, we get some resources, but this is not always enough. For example, we will not be able to access pages we do not have permission to view, or there will be some pages in the application interface that we would not see visually. So, what are the alternatives?

Taking advantage of web application history (Wayback Machine)

As you know, archive.org holds archives of web applications.

Although the web application we view has changed over time, some of the old pages may still be there. The information we get from here can sometimes make it easier for you than you can imagine.

It is possible to find some tools in Github that will allow you to search more easily in the archives here.

Obtaining full URL and relative path information from Javascript files

The more content we discover on the website/server that we analyze, the more likely we are to find vulnerabilities.

We may also be able to access a misconfigured service, the admin panel, or the debug service, which may be forgotten and open to access by default passwords.

Since functions such as file uploading or command execution are often required by design, there may be no need to look for further weaknesses.

relative-url-extractor is a tool that can be used directly in both local and remote Javascript files¹.

```
mg@Netsparker-VirtualBox:~/relative-url-extractor$ ruby extract.rb https://www2.assets.
/
/a
//
/events/
/log
/contact-sales
/html/continuous-delivery/ci-animation.html
/html/continuous-delivery/ci-flow-animation.html
/html/dynos/dyno-build.html
/event_tags.json
/html/kafka/kafka.html
/html/opex/opex-diagram.html
/html/platform-scale/platform-scale.html
/html/spaces/spaces.html
//cdn.jsdelivr.net/algoliasearch/2/
/1/indexes/
/1/logs?
/1/indexes
/1/keys
/1/keys/
/1/places/query
```

LinkFinder²

```
python linkfinder.py -i http://angular.testsparker.com -d -o cli
```

```
mg@Netsparker-VirtualBox:~/relative-url-extractor/LinkFinder$ python linkfinder.py -l http://angular.testsparker.com -d -o cli
https://fonts.googleapis.com/icon?family=Material+Icons
https://fonts.googleapis.com/css?family=Roboto:300,400,500,700,400italic
```

Identifying vulnerable JavaScript libraries

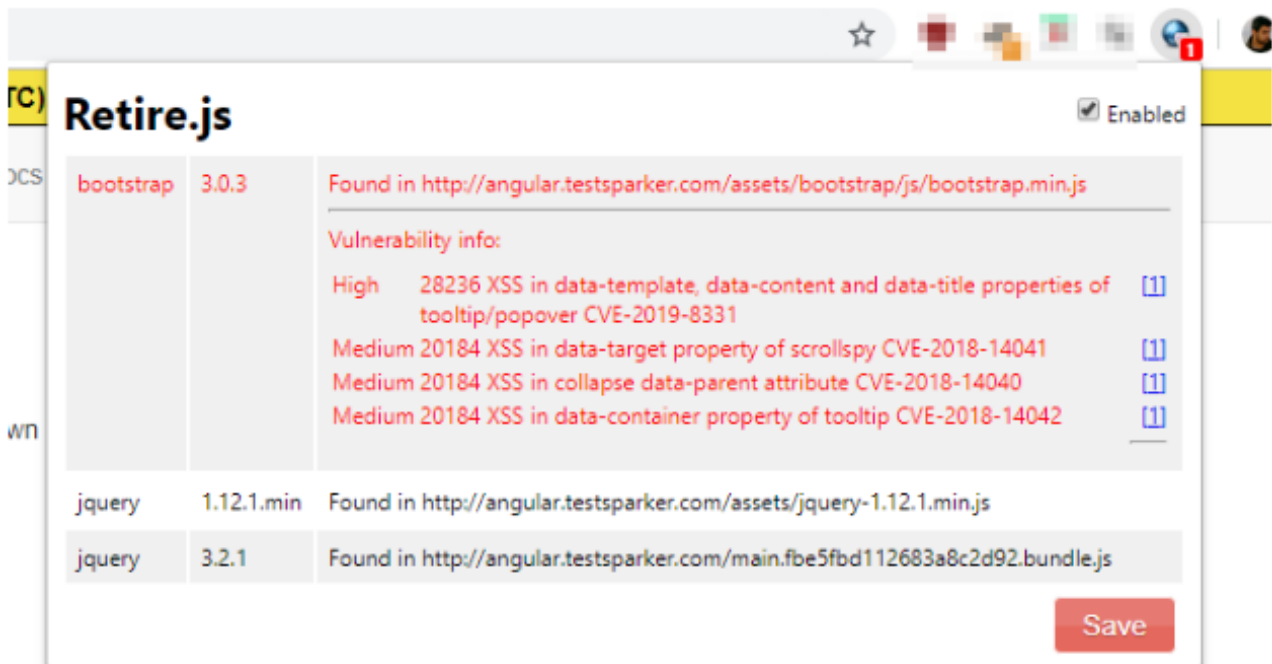
The application you are testing may be using a library that a vulnerability has been discovered within. If this is the case, it may be possible to exploit the previously identified vulnerability on the application we tested. So how do we check this?

Retire.js³ will meet all of our needs here. You can install Chrome and Firefox as a browser plug-in. You will be greeted by a screen like the one below when visiting the sites where it was established.

1 <https://github.com/jobertabma/relative-url-extractor>

2 <https://github.com/GerbenJavado/LinkFinder>

3 <https://github.com/RetireJS/retire.js>



Using directly through the browser is often impractical because it requires browsing the pages. In such cases, it is often effective and easy to use from the command line.

```

ng@Netsparker-VirtualBox:~$ retire -h

Usage: retire [options]

Options:
  -h, --help                output usage information
  -V, --version             output the version number

  -p, --package             limit node scan to packages where parent is mentioned in package
  -n, --node                Run node dependency scan only
  -j, --js                  Run scan of JavaScript files only
  -v, --verbose             Show identified files (by default only vulnerable files are shown)
  -x, --dropexternal        Don't include project provided vulnerability repository
  -c, --nocache             Don't use local cache

  --jspath <path>          Folder to scan for javascript files
  --nodepath <path>        Folder to scan for node files
  --path <path>            Folder to scan for both
  --jsrepo <path|url>      Local or internal version of repo
  --noderepo <path|url>    Local or internal version of repo
  --cachedir <path>        Path to use for local cache instead of /tmp/.retire-cache
  --proxy <url>            Proxy url (http://some.sever:8080)
  --outputformat <format> Valid formats: text, json, jsonsimple, depcheck (experimental) and
  --outputpath <path>      File to which output should be written
  --ignore <paths>         Comma delimited list of paths to ignore
  --ignorefile <path>      Custom ignore file, defaults to .retireignore / .retireignore.js
  --severity <level>       Specify the bug severity level from which the process fails. All
  --exitwith <code>        Custom exit code (default: 13) when vulnerabilities are found
  --colors                 Enable color output (console output only)
  
```

There is a comprehensive list of which libraries can be detected at <https://retirejs.github.io/retire.js/>, which can be checked in detail.

Synk⁴ is another tool that we can use for similar purposes.

You will often find false positive findings when using these tools,

Extracting critical information such as API keys, user passwords

Other critical information that we can find in Javascript files is the data we can use directly in the system, such as forgotten API keys and user passwords.

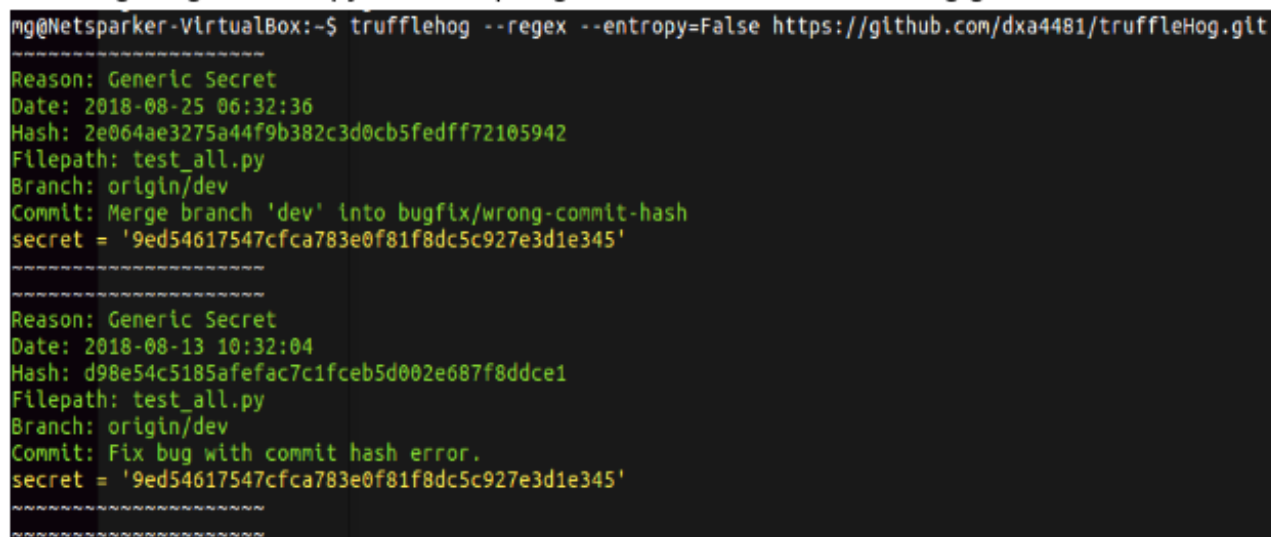
At this point, there are a few basic approaches followed by tools that can help us. One of them is Regex based search and matching tools. These tools search with the regex logic we know. They can determine the data to be accessed by the rules entered in the store/directory/page entered as targets. These tools are often successful in detecting user names and passwords.

The other method is tools that make Entropy-based searches. These tools are generally successful in finding data such as API key and token.

Both regex and entropy-based tools are often able to deliver FPs.

truffleHog: <https://github.com/dxa4481/truffleHog>

```
trufflehog --regex --entropy=False https://github.com/dxa4481/truffleHog.git
```



```
mg@Netsparker-VirtualBox:~$ trufflehog --regex --entropy=False https://github.com/dxa4481/truffleHog.git
Reason: Generic Secret
Date: 2018-08-25 06:32:36
Hash: 2e064ae3275a44f9b382c3d0cb5fedff72105942
Filepath: test_all.py
Branch: origin/dev
Commit: Merge branch 'dev' into bugfix/wrong-commit-hash
secret = '9ed54617547cfca783e0f81f8dc5c927e3d1e345'

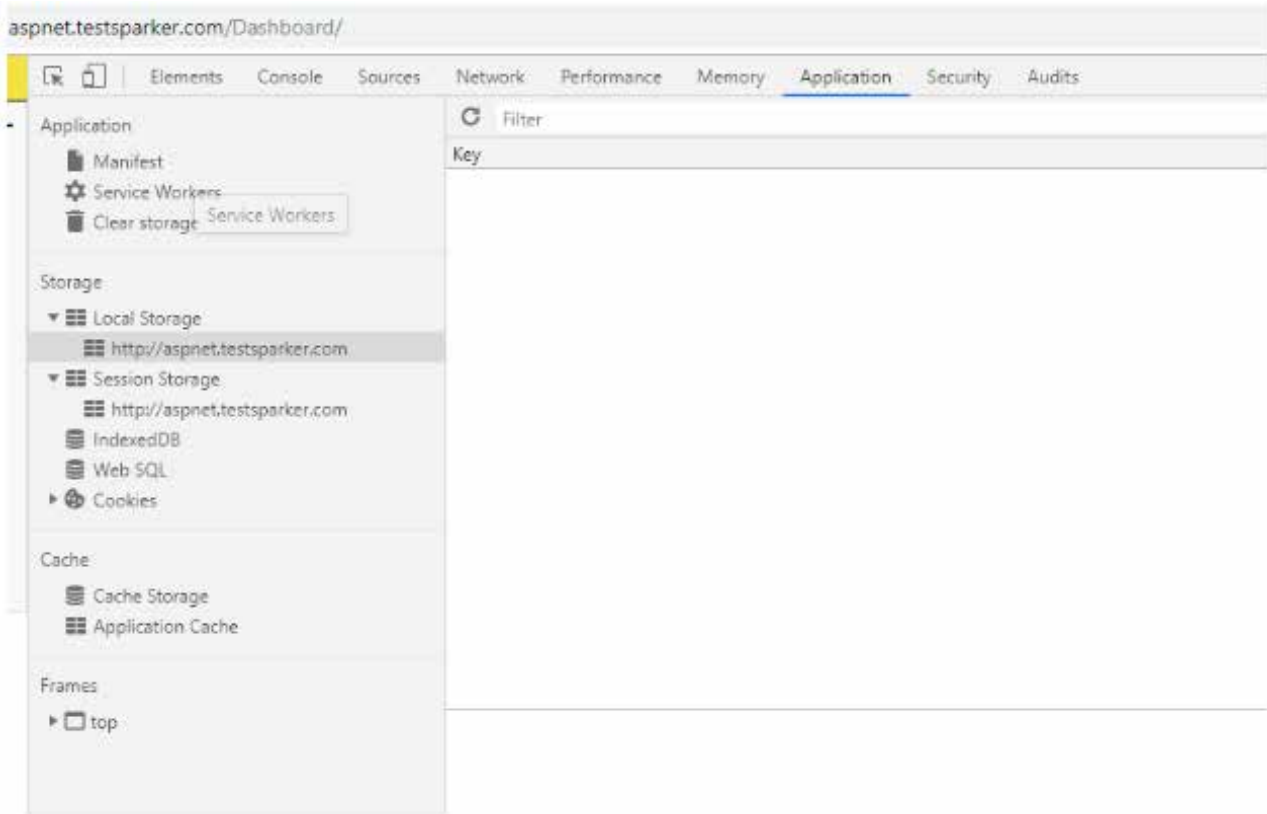
Reason: Generic Secret
Date: 2018-08-13 10:32:04
Hash: d98e54c5185afefac7c1fceb5d002e687f8ddce1
Filepath: test_all.py
Branch: origin/dev
Commit: Fix bug with commit hash error.
secret = '9ed54617547cfca783e0f81f8dc5c927e3d1e345'
```

Cookies and other data in the browser

Browser Storage is another place where we can access information that may make sense to us. In DevTools, we can easily access these files via the Application tab.

⁴ <https://snyk.io/>

It is possible to store information in two different locations. These are Local Storage and Session Storage. The main difference between them is the data stored in Session Storage is deleted when communication with the application is interrupted when the browser itself or the corresponding browser tab is closed. If no special condition is defined for data stored in Local Storage, it can be accessed until it is deleted.



References and Sources:

- [Performing JavaScript Static Analysis by Lewis Arden \[Video\]](#)
- https://medium.com/@_bl4de/how-to-perform-the-static-analysis-of-website-source-code-with-the-browser-the-beginners-bug-d674828c8d9a

Encrypt Your Disks with BitLocker

Why Drive Encryption?

As you know, there are passwords we use to protect our computers. If these passwords are not entered, you cannot log into the operating systems. Are you one of those who think that these worries are nonsense?

Okay, but what if someone who has physical access to your computer, such as someone stealing your laptop, directly accesses your HDD where important data is stored - will these trusted passwords protect you? Of course no!

You don't always need to be away from your laptop for physical access. As mentioned in our article on the Haven application developed by Edward Snowden, you may be the subject of an attack called Evil Maid, even when you leave the hotel room you left for lunch. So, someone dressed as a housekeeper can get into your room and get physical access to your PC.

As a precaution, we must encrypt our discs. In this article, we will talk about how to enable BitLocker on Windows, which allows us to encrypt our disks.

What is BitLocker?

BitLocker Drive Encryption is a local security feature that encrypts everything on the drive on which it is enabled. Device encryption helps protect your data by encrypting it. Only a person with the correct encryption key (such as a personal identification number) can decrypt data on the device.

How does it work?

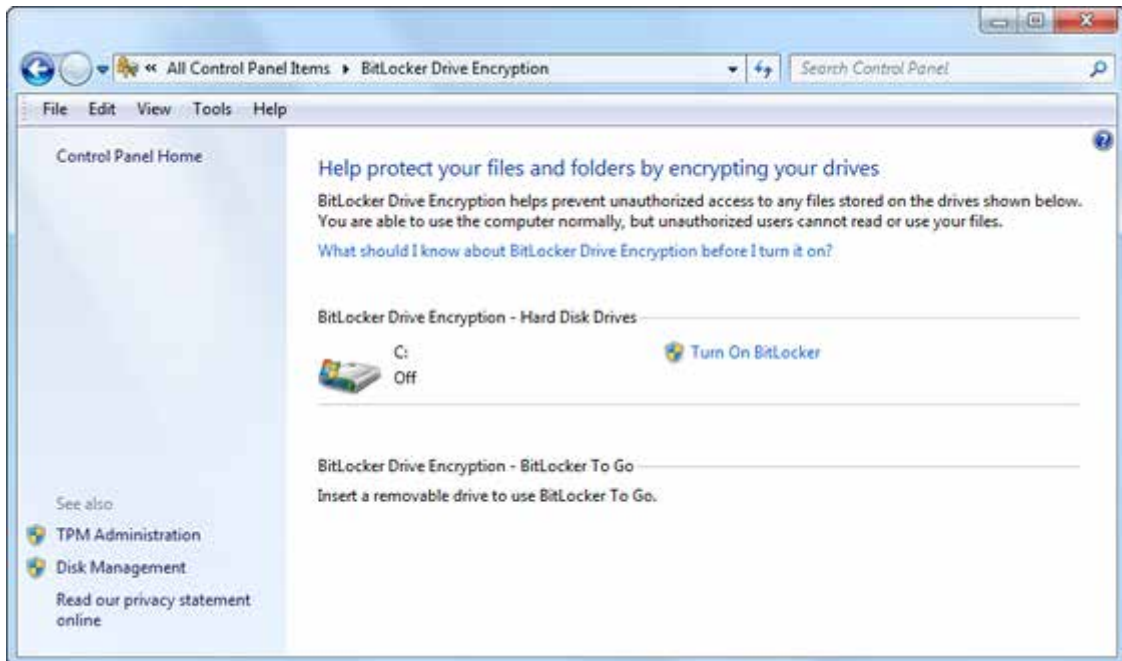
BitLocker is used with a hardware component called the Trusted Platform Module (TPM). TPM is a chip on the motherboard. BitLocker stores the recovery key in the TPM (version 1.2 or later).

When you enable BitLocker, you must enter a Personal Identification Number (PIN) every time you start your computer. When enabling BitLocker a recovery key is generated. This recovery key is used to access your computer if you forget your password. After the recovery key is generated, you are prompted to restart the machine. The encryption process starts when the computer restarts.

Activating BitLocker

1- Click Start -> Control Panel -> System and Security. After these steps, you should see "BitLocker Drive Encryption. Please click "BitLocker Drive Encryption. Note: BitLocker is not available in all versions of Windows.

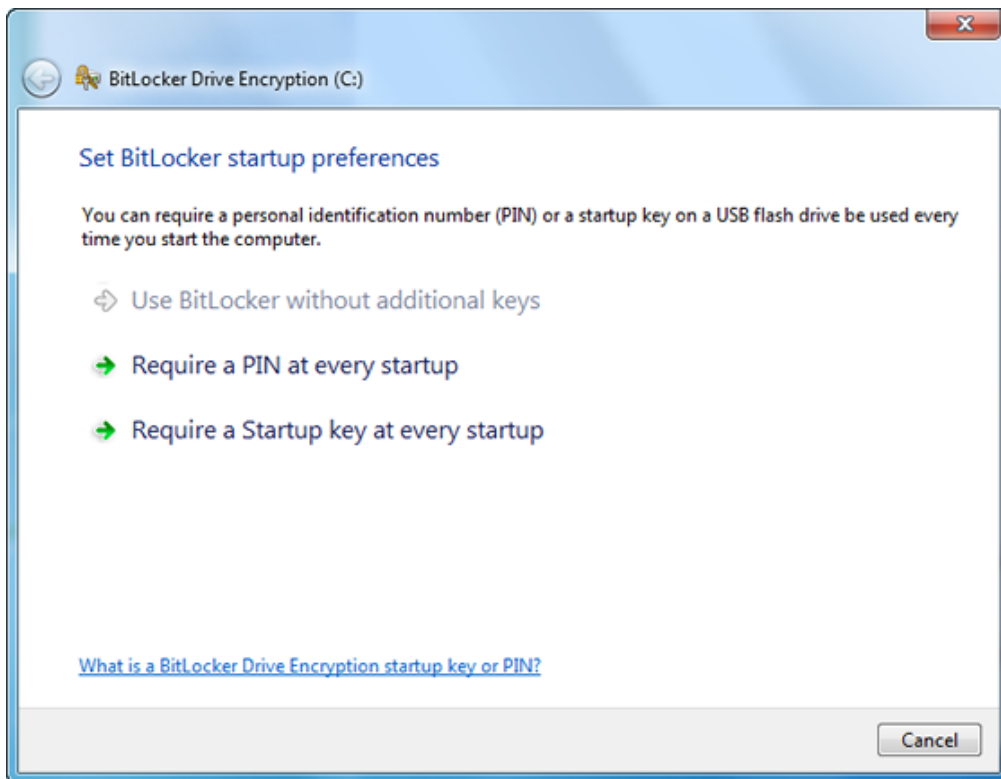
2- Click “Turn on BitLocker”



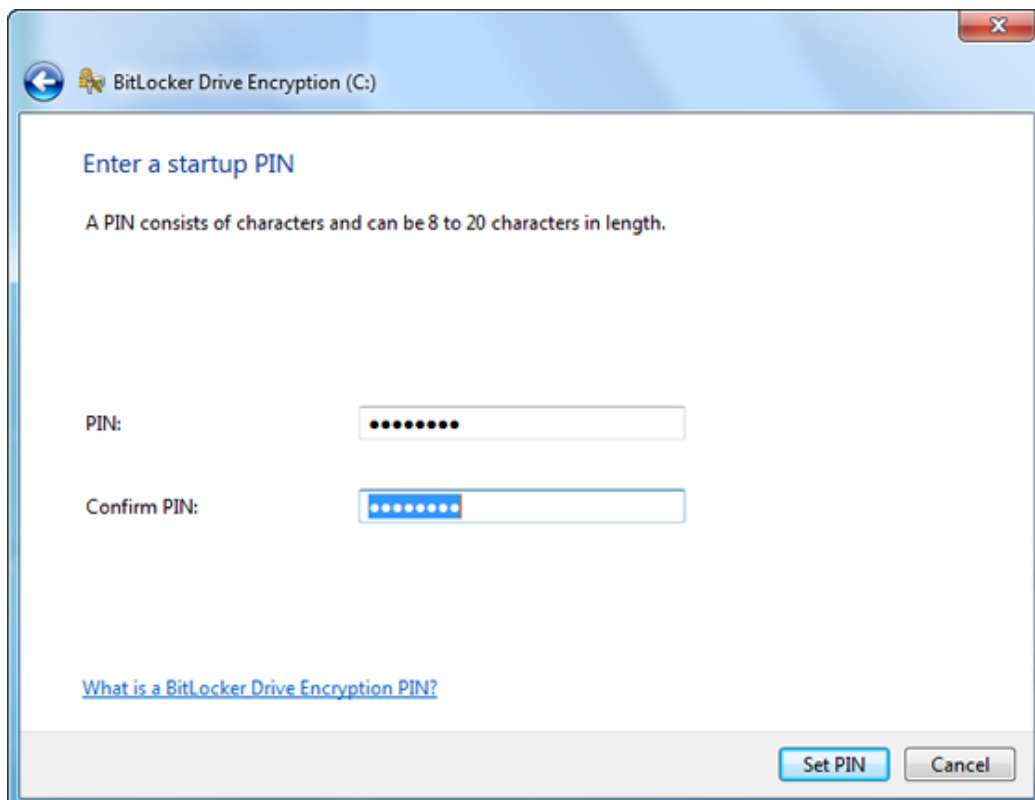
3- BitLocker will scan your computer to verify that it meets the system requirements. If you do not encounter any problems, please click “Next”.



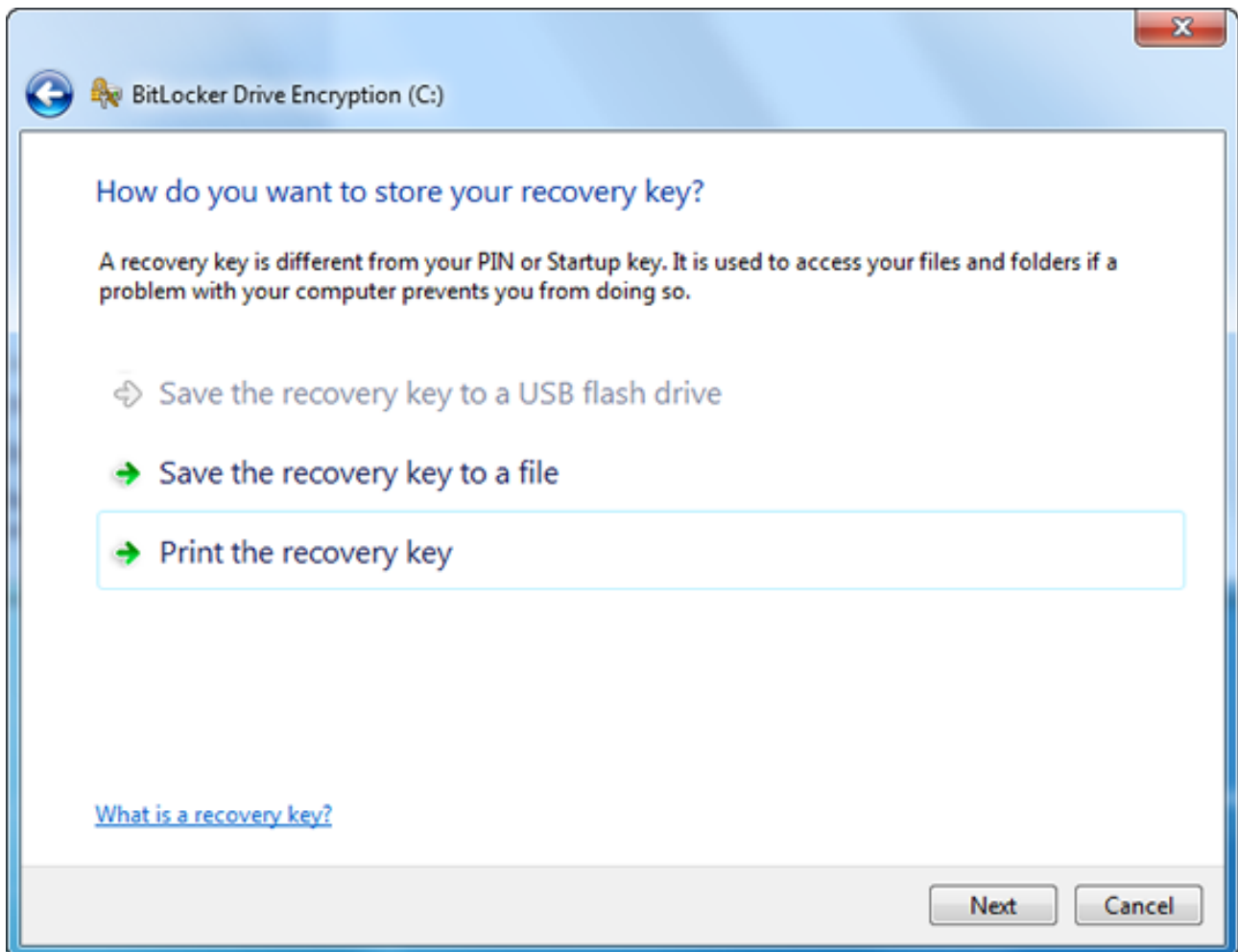
4- When the “BitLocker startup preferences” page is displayed, click ”Require a PIN at every startup”.



5- Enter a PIN which is between 8 to 20 characters in length, and then re-enter it in the “Confirm PIN” field. Click “Set PIN”.



6- To save your recovery key, select "Print the recovery key" and click "Next"



A pop-up will be prompted telling to restart your computer to start the encryption process. After your computer restarts, you can safely use your computer!

OLDEST OF THE HACKERS II - RICHARD GREENBLATT

“Hackers like Richard Greenblatt, Bill Gosper, Lee Felsenstein, and John Harris are the spirit and soul of computing itself. I believe their story their vision, their intimacy with the machine itself, their experiences inside their peculiar world, and their sometimes dramatic, sometimes absurd “interfaces” with the outside world is the real story of the computer revolution.”¹

Hello everyone, in our previous article in the third issue of Arka Kapi Magazine, we mentioned MIT, where the concept of hacker was born, AI Lab (Artificial Intelligence Lab), TMRC (Tech Model Railroad Club), computers manufactured by the DEC (Digital Equipment Corporation) such as PDP-1, PDP-6 and TX-0. We talked about laying the foundations of the hacking culture (especially those who were members of the TMRC’s Signal and Power Subcommittee, which later turned their paths to programming), and the life of Bill Gosper, one of the rare people who laid these foundations. Let’s cover those a little to help you remember. Those who have been or will be mentioned (Bill Gosper, Richard Greenblatt, Alan Kotok, Peter Samson and many others) have made history with the extraordinary things they have done starting from the 50-60s; they’re the people who developed the idea of hacking and hacker ethics; these people are the real hackers who gave birth to the concept of hacking. In short, hacking and hacker concepts can be explained as follows: “thus, hacking means exploring the limits of what is possible, in a spirit of playful cleverness. Activities that display playful cleverness have *hack value*.”² In this article, we are going to be talking about a legendary hacker, a chess master and an amateur radio fan, whose name is often mentioned with Gosper. Ladies and gentlemen, I present to you: Richard Greenblatt!

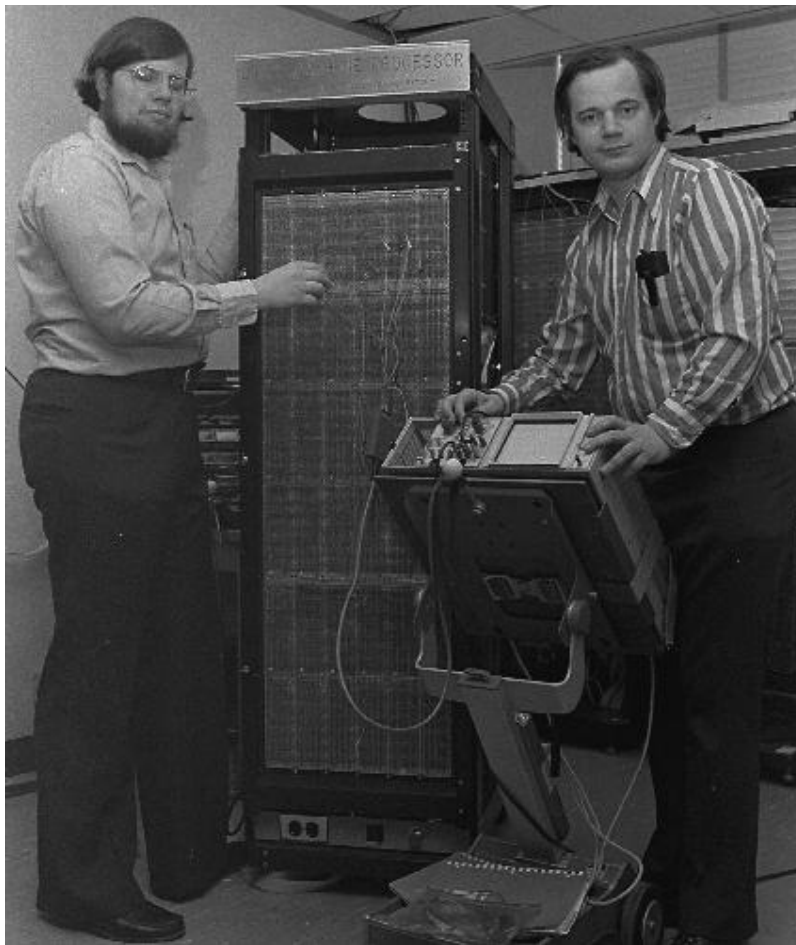
Richard D. Greenblatt was born on December 25, 1944, in Portland, Oregon, and a few years later after his parents divorced, he moved to Columbia, Missouri with his mother and sister. Columbia is where Greenblatt grew up. It is a college town located between St. Louis and Kansas City. At that time, half of the city’s population was students. According to Greenblatt, this is an ideal place for growing up. When his father came to visit Columbia from Philadelphia, he took 9-year-old Greenblatt to the University of Missouri Student Union, where Greenblatt found such intelligence that he could not find within his peers; he was particularly good at chess, and he did not even have trouble defeating the college students there. One of the people he defeated in chess, Lester, a University of Missouri student, gave Greenblatt a gift to immerse him in the electronic world. The duo later worked together on the production of amateur radio (aka ham radio). At the end of his university, Lester introduced Greenblatt to Mr Houghton who ran a local radio shop, and this place became Greenblatt’s second home during high school. Here, Greenblatt and his friend did a lot of projects from amplifiers to modulators, oscilloscopes and raw radios. Finally, high school ended there came the time to choose a university.

1 Steven Levy, HACKERS: Heroes of the Computer Revolution

2 <https://stallman.org/articles/on-hacking.html>

In the fall of 1962, Richard Greenblatt enrolled in MIT. Since he handled the challenging first-year courses, he was among the lucky ones to take EE 641 - *the Introduction to Programming* elective course - hereby, he often went to the punch-card machines to write programs to the IBM 7090. At the same time, his roommate Mike Beeler, was also taking the Nomography course, and Greenblatt used to accompany Beeler on his way to the IBM 1620: where you'd wait in line for a punching you card stacks, and when was your turn, you could put those cards into the reader and see the output instantly. Beeler and Greenblatt felt a great passion people felt when they did other things, like watching a game or drinking beer with friends. In December 1962, when it was almost Christmas time, Greenblatt became comfortable around the TMRC. There, among all the giant layouts in the TMRC, *Peter Samson* wrote a big timetable program for the TMRC operating sessions in FORTRAN for the IBM 7090. In order to make it possible to write this timetable the program also on PDP-1, Greenblatt developed a PDP-1 FORTRAN compiler. There is no special reason he attempted such a job; "if you want a machine to do a job, but the machine does not have the software to do it, you write the software that you need, so that it can be done".

In his first year at university, Richard Greenblatt had already become an outstanding student without much effort. In the sophomore year, Greenblatt started working at the university with a salary below the minimum wage. Like a few other hackers, he was working on developing great programs that were written on the system or on artificial intelligence.



3

Richard Greenblatt with Thomas Knight

3 <https://www.computerhistory.org/chess/stl-431614f64ea3e/>

MIT and some corporate customers were moving to the Tech Square - a hacking monastery on the Main Street - and alongside with them, a Project MAC and a second PDP-1 were being moving to as well. The computers were on the 9th floor here, and guess who was the most time-consuming hacker? Bingo! Richard Greenblatt! Greenblatt was hacking very intensely; he worked nonstop for 30 hours, after where there was no sign of energy left, he slept for 12 hours at home or right at the lab. This *30-hour-day* of course did not fit into other scheduled tasks such as meetings, exams or lectures. Greenblatt was so detached that his mother came to Columbia from MIT (which is about 20 hours by car, given the circumstances of that time) to discuss the status of his son with the dean. As far as Greenblatt's roommate Beeler mentioned, we can know that his mother was very worried. However, for Greenblatt, this didn't mean anything. Although it was true that his school - his *graduation* was in danger; after all, hacking was the pinnacle of everything for him, and nothing else would make him happy that much. As Steven Levy states in *HACKERS: Heroes of the Computer Revolution*; "*When you encounter something else that puts lessons elsewhere in your eyes, you can throw them aside.*" That's exactly what happened to Greenblatt. The topic was the hacking itself. Not only Greenblatt, but also anyone who spent time at the beginning of the TMRC or PDP-1 thought that "hacking is such a satisfying quest that it can even be a way of life," and it was quite consistent. One day, when he couldn't take a final exam because he overslept, and let the chips fall where they may: Greenblatt was no longer a student at MIT.

*"...hacking not only had an understanding of the system, but also a creative control of dependency and the illusion that full control was only a few features away."*⁴

Thus, Greenblatt started looking for work: something that would enable him to develop a program during the day and spend time on Tech Square's 9th floor (by computers) at night. As a matter of fact, he found just what he wanted. Richard Greenblatt began working at Charles Adams Associates. Greenblatt was in Boston all day long on the Highway Technology Highway şehir in the countryside, and at night at MIT (long live hacking at night!) 48 kilometers away. When Charles Adams Associates was terminated, he was hired again by MIT AI Lab. Although he lived as a pensioner in a retired dentist's home in Belmont, he usually slept on a portable cot on the 9th floor of Tech Square. Clearly, Greenblatt's priorities were different: hygiene, for example, or taking a bath, were insignificant. For the same reason, he has been expelled from the dormitories at Cambridge YCMA because he could not keep his room clean. Greenblatt was totally committed to hacking. His speech pattern was strange, he sounded as if he was constantly mumbling and when he tried to speak properly, caused nothing but wobbling. Other hackers like Gosper, Kotok, and Samson had adopted these features of Greenblatt, which, according to them, had a cute hilariousness. As Gosper said: "He was a complete pragmatist. What people thought, be damned. If anyone thought he was stupid or nerdy, that was their problem. Some people did, and they were wrong..." Speaking of Gosper, it would be nice to mention a little bit about Greenblatt's connection with him.

Bill Gosper and Richard Greenblatt. Two hackers, two masters. Bill Gosper: a mathematical genius who could speak better than Greenblatt, who apparently didn't talk much properly. This legendary duo, which respected each other's strengths and different aspects, represented two different types of hacking: while Gosper worked on mathematical discovery, Greenblatt concentrated on developing pragmatic systems. Both were involved in projects where everyone's best side was used - generally working together. We would like to thank this duo for watering and diligently raising the Hacker Ethic, a precious flower that sprouted on the 9th floor of the Tech Square! Hacker Ethic was able to advance to the top thanks to this magnificent nest of technology. Here, at the TMRC, there were heated discussions about finding *The Right Thing*. We mentioned them in the previous article, but let's cover it briefly. These discussions were invaluable: for example, Alan Kotok, a DEC employee, was often present at the Tool Room, even making important decisions about the design of the PDP-6 during the discussions there. Both Gosper and Greenblatt had strong arguments. However, Greenblatt would soon get tired of dealing with corrosive human interface and start doing - scribbling something, or sitting in front of the PDP-1 console and screaming the code. Elegant, or not: according to Greenblatt, things had to be done. The programs he wrote had an amazing built-in error check: robust and

⁴ Steven Levy, *HACKERS: Heroes of the Computer Revolution*

firmly debugged when finished. They were so finely debugged that Gosper even thought that Greenblatt occasionally wrote bug codes just to debug them.

Later, Greenblatt and a few hackers helped Alan Kotok compile LISP, John McCarthy's artificial intelligence language, in PDP-6. After some time, the blackboards in the TMRC were filled with lines of code and finally the compiler was running on the machine! Hackers started to integrate this MacLISP language into their programs, even in their lives. Take the 'P' used for suggestions, for example. Instead of "Shall we eat?": eat-P⁵ meant the same thing to them. Lisp probably didn't put Assembly's nose out of joint, but to Gosper and Greenblatt: Lisp was the perfect fit for Hacker Ethic.

Greenblatt had been a great chess player his whole life. He was also a legendary hacker. So why not merge chess and hacking? Thus, he wrote a chess program using sophisticated artificial intelligence techniques which tried and figured out moves in accordance with certain criteria Greenblatt considered good chess. The program was able to compete against a real player within a week. Within the next few months, it was debugged, some features added and Mac Hack came to life! "Artificial Intelligence Skeptic" Hubert Dreyfus argued that computers could not make a quality chess game and could not beat even a 10-year-old boy (let's not forget that Greenblatt was 9 when he defeated the college students). So, MIT invited Dreyfus to play chess against Mac Hack on PDP-6. Result: checkmate! Dreyfus was defeated badly⁵. Peter Samson explains the moment when Dreyfus was defeated: "the defeated critic returned to the professors and looked at the hacker team, including the program creator Greenblatt, but did not see any enthusiasm; no applause, no celebration. Because they knew... Dreyfus was part of that real world that couldn't understand the fascinating nature of computers. It was not as interesting to have skeptical people believe in hacker ethics as it is to live in hacker ethics. Those in the real world wouldn't understand how the 30-hour-day dedication of the Assembly sect, discovering stuff being comfortable around computers, changing the world for the better and the experiences of Gospers and Greenblatts felt like; how it was to *be a hacker*. In 1977, the undefeated Bobby Fischer played three games against Greenblatt's program and beat all three.

In 1979 Greenblatt became the main developer of MIT Lisp Machine with Tom Knight and founded a company called Lisp Machines, Inc. (which was later renamed as Gigamos systems). Later, along with Tom Knight and Stewart Nelson, he was involved in writing the *Incompatible Timesharing System*, an extremely efficient time-sharing operating system for PDP-6 and PDP-10.

"As I talked to these digital explorers, ranging from those who tamed multimillion-dollar machines in the 1950s to contemporary young wizards who mastered computers in their suburban bedrooms, I found a common element, a common philosophy which seemed tied to the elegantly flowing logic of the computer itself. It was a philosophy of sharing, openness, decentralization, and getting your hands on machines at any cost to improve the machines, and to improve the world. This Hacker Ethic is their gift to us: something with value even to those of us with no interest at all in computers."

-Steven Levy

Source :

<http://www.computerhistory.org/collections/catalog/102657935>

[https://en.wikipedia.org/wiki/Richard_Greenblatt_\(programmer\)](https://en.wikipedia.org/wiki/Richard_Greenblatt_(programmer))

<https://www.ithistory.org/honor-roll/mr-richard-d-greenblatt>

⁵ <https://ingram-braun.net/public/research/parlour-games/article/computer-chess-richard-greenblatt-match-mit-philosophy-artificial-intelligence-history/>

Our Fellow Confidant, RSA

*“The enemy knows the system”
Claude Shannon*

What is RSA and what isn't? No, it is not a code block of a programming language. RSA are equations based on Leonhard Euler's theorem¹, expressed in the language of mathematics. We the developers are the intermediaries translating equations written in mathematics language to computer language. We again need to express it mathematically when cracking the RSA algorithm, attacking, and searching for back doors. During the translation into computer language, weaknesses arising from the software developer, software and hardware do not harm RSA's reputation. You can implement the RSA algorithm with any software language such as QBasic, C #, Java, Python. The reliability of an algorithm is completely independent of the software code.

However, this article is not about the history of RSA - there are enough resources available for that. In this article, we are going to look at RSA's theory, maths and applications. By the cryptology science, RSA is classified in the asymmetric encryption class. It has been created in 1977 by Ron Rivest, Adi Shamir and Leonard Adleman. Named after the initials of the creators.

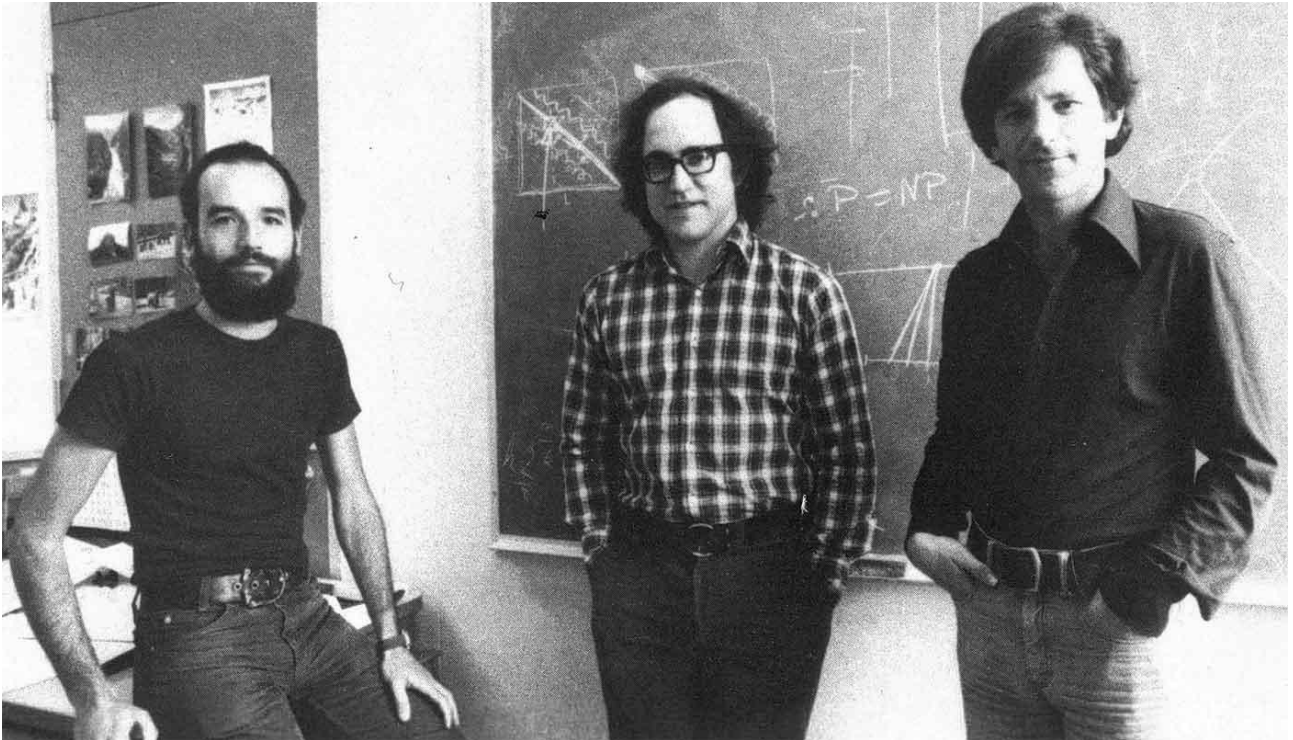


Figure 1: from left to right, Adi Shamir, Ron Rivest, Leonard Adleman

¹ www.wikipedia.org , “Leonhard Euler” , https://tr.wikipedia.org/wiki/Leonhard_Euler

Proof of RSA Algorithm according to Euler Theorem

What does RSA algorithm's mathematical language say, how is it proven? Halit İnce, one of the Arka Kapi Dergi writers, answered these questions for us. If you say that the Mathematical language of the algorithm is not necessary for you now, you can continue without reading this proof.

Euler's Theorem: Let a be any integer and n be a prime integer in between a .

Then, $a^{\varphi(n)} = 1 \pmod{n}$ equality is provided. In the following sections of the article, c will be representing cipher text, m open text, n key will represent n phi number e and d numbers will be the exponent values

Why does c^d equal m ?

m is found by performing $c^d = (m^e)^d = m^{ed} = m^{1+k\varphi(n)} = m \cdot (m^{\varphi(n)})^k = m \cdot 1^k = m \pmod{n}$

There are 6 equalities. Let's explain each of them.

1st Equation: Here, only m^e is written instead of c . Remember that $c = m^e \pmod{n}$.

2nd Equation: Is apparent.

3rd Equation: Since $ed = 1 \pmod{\varphi(n)}$, the remainder of the division of ed to $\varphi(n)$ is 1, and the division is any integer, such as k . So, $ed = k\varphi(n) + 1$. We used this in the equation. d was chosen so at the beginning to provide such equality.

4th Equation: Exponential written again.

5th Equation: There are two cases:

Case 1: If m and n are coprime integers, from Euler's Theorem, it is proven that $m^{\varphi(n)} = 1 \pmod{n}$ and in the 4th equation, 1 is written instead of $m^{\varphi(n)}$. We can use the theorem since m and n coprime integers.

Case 2: If m and n are not coprime integers, then:

$$m \cdot (m^{\varphi(n)})^k = m \cdot (m^{(p-1)(q-1)})^k = m \cdot (m^{(p-1)})^{(q-1)k} = m \cdot (1)^{(q-1)k} \pmod{p}$$

$m^{(p-1)} = 1 \pmod{p}$ (Euler's Theorem) was used in the last equation. Observe that $\varphi(p) = p-1$ and that m and p are coprime integers.

$$m \cdot (m^{\varphi(n)})^k = m \cdot (m^{(p-1)(q-1)})^k = m \cdot (m^{(p-1)})^{(q-1)k} = m \cdot (1)^{(q-1)k} \pmod{q}$$

$m^{(p-1)} = 1 \pmod{q}$ (Euler's Theorem) was used in the last equation. Observe that $\varphi(q) = q-1$ and that m and q are coprime integers.

Since, in this case, the expression $m \cdot (m^{\varphi(n)})^k$ is 1 relative to both \pmod{p} and \pmod{q} ;

Then, $m \cdot (m^{\varphi(n)})^k = 1 \pmod{pq} = 1 \pmod{n}$.

Hereby, in both cases, we can write 1 in place of $m^{\varphi(n)} \pmod{n}$ in the 5th equation.

6th Equation: Apparent.

So, the result of the operation $c^d \pmod{n}$ will always give the m open text.

After taking a look at the proof, let's resume the article in its own course. In the application, we can examine the RSA algorithm in two stages. The first stage is the installation process: keys to be used continuously in the system are created here. These operations are performed once and are not repeated as long as the keys created are in use. The second stage is where these keys are used in encryption and decryption operations, that is to say, routine and repetitive operations.

A- CONFIGURATION OF THE SYSTEM

Consist of 5 steps that need to be done painstakingly.

1- The first step to generating a private and public key in the RSA algorithm is to generate two random numbers. There are strict rules that must be applied while selecting these numbers. Because RSA algorithm which ensures the confidentiality of the whole communication and data is secure thanks to these strict rules.

Let the numbers to be created be p and q .

- p and q must be chosen randomly! If you relate these two numbers in any way, the attackers would benefit from it. It is recommended to read the When Random Numbers Aren't Random article written by Chris Stephenson, published in Arka Kapi Magazine's 5 issue, to gain further information on random number generation and how the link between the number generated can be dangerous.
- p and q numbers must be prime. Mathematics used in the RSA algorithm has been based on the features of prime numbers. It is a must. We need to verify that each and numbers are prime. So-called secure RSA applications which use and non-prime numbers should be avoided. There are researches in which some application have overseen this fact. That is because applying prime number test to 1024, 2048, and 4096 bit numbers require a high hardware and time cost. One of the oldest known methods used in prime number testing is the Sieve of Eratosthenes ². This method which is beneficial while working with small numbers would not promise high performance for larger numbers. That'd be bulky. As the numbers get larger, a solution needs to be found against serious performance problems encountered during prime number tests. No commercial application would want to show the user the loading icon more than a few seconds. This is why during prime number tests, rather a performance based approach is followed for large numbers. As the number of digits increase, methods who can say that a number is certainly not prime, yet can not describe that it is prime. In our application, since we are operating with small numbers, Sieve of Eratosthenes which produces exact results will be used. You can find details inside the application.

Prime Numbers ³

We all know that prime numbers are such integers as they can only be divided by themselves and 1. Prime numbers are known as the atoms of numbers. They are the non-interactive, asocial numbers through the whole set of infinite numbers. They resemble mysterious people who live in their houses without interfering with the public. There are all sorts of theories about them. There are infinite numbers of prime numbers, like {2,3,5,7,11,13,17,19...} .

- Numbers p and q must be close by means of digit number. The only way to solve the RSA algorithm is to factorize the generated keys. This is the only we know for now. To improve the security and robustness of the encryption method, we should take every opportunity, even as much as crumb.

² https://en.wikipedia.org/wiki/Sieve_of_Eratosthenes

³ https://en.wikipedia.org/wiki/Prime_number

Prime Factorization

The numbers that can be written as the product of two prime numbers are called semi-prime numbers. And all the remaining positive numbers from the prime numbers are called composite numbers. Composite numbers can be expressed as the product of integers smaller than them. Prime factorization is finding the prime numbers that give the original number when multiplied. As the numbers get larger, factorization gets more difficult. Prime factoring two close numbers is much more difficult. Factoring random, close and quite large (say, 1024 bits) semi-prime numbers is not pretty possible with today's technology. At least, not practical. An algorithm to speed up the factoring process has not yet been found as well. In 2009, a group of researchers factorized a 768-bit number in 2 years, using hundreds of computers.⁴

- Numbers p and q should also be not very close. As we will see further, the key, say number n , is obtained using the simple and neat expression $n = p \times q$. You can be sure that the method the attacker will try first will be square rooting, i.e., finding the \sqrt{n} value and search for p and q somewhere near this value. For this reason, p and q should never be generated having close-values. Considering the technological possibilities of the day, the gap between the two numbers should be large enough to neutralize brute-force attacks.
- There should be no relation nor pattern between the numbers p and q . Suppose a number set given as $\{0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, \dots\}$. One can immediately recognize them as the Fibonacci numbers since every number in the Fibonacci series is equal to the sum of two consecutive numbers before it. Fibonacci numbers can be calculated according to specific rules. A proven rule similar to Fibonacci numbers for prime numbers has not yet been found. If you are to find this rule, regardless of your age, you will be given the next few Field Medals⁵. Thanks to you, RSA would be thrown into the trash and this article would lose its value. Yet, there are some researches like the Riemann hypothesis⁶ who are taken seriously but have not yet proven.
- Numbers p and q are private and aresh never to be shared!

2- Now that we have generated the p and q numbers by fulfilling all strict rules, we can proceed to generate the encryption key. With the number being a part of both the public and private key, $n = p \times q$. By using something as fundamental as primary school multiplication, we have calculated a crucial part of the RSA key. Only, the numbers are too large! The bit length of the number n is given as the key digit length (bit in binary) of the RSA algorithm. This number n is used as the modular base for the private key and the public key. Now, you might be wondering why we struggled so much to generate the numbers p, q - at the end we could have directly generated a much larger value. There are many reasons for this struggle. Let's explain:

The Fundamental Theorem of Arithmetic⁷

The Fundamental Theorem of Arithmetic states that, any natural number greater than 1 can be written as the product of a finite number of prime numbers. If the displacement of the numbers is not taken into account, then this multiplication is unique. Explaining simply: let the numbers a and b be prime. Except for $c = a \times b$ or $c = b \times a$ sequence, there exists no group of prime numbers as to multiply the number of c .⁸

4 "Factorization of a 768-bit RSA modulus", <https://eprint.iacr.org/2010/006.pdf>

5 https://en.wikipedia.org/wiki/Fields_Medal

6 https://en.wikipedia.org/wiki/Riemann_hypothesis

7 https://en.wikipedia.org/wiki/Twin_prime

8 <https://www.khanacademy.org/computing/computer-science/cryptography/modern-crypt/v/the-fundamental-theorem-of-arithmetic-1>

Since we are the ones that determined the prime numbers p, q , it is also us who determined number n 's prime factors. We are forcing the attacker to find two prime numbers, which we generated with strict rules. As the idiom goes, we're forcing the attacker to search for a needle in a haystack. Because the number n is public and is the heart of the RSA algorithm. If the foe knows the system, you need to protect the key very well. In my opinion, it is a good sign to be paranoiac in the field of Cryptology, it shows that you are doing your job meticulously.

Another benefit of having generated the prime numbers p, q ourselves is that in the next step it is totient - in short, makes it easier to calculate the number $\varphi(n)$. If we generated the number n directly, we would have a hard time generating the number $\varphi(n)$. In order to find the number $\varphi(n)$, we would have to check whether each integer from 1 to n is mutually prime with n . This would be a challenging software process, such as prime factoring a 300-digit number.

3- Now that we have calculated the number n , it is time to find the value of $\varphi(n)$. The number $\varphi(n)$ is a private number, should never be shared.

Totient⁹, shortly φ , is a function in number theory which counts the positive integers up to a given integer n that are relatively prime to n . Developed by Swiss mathematician Leonhard Euler.

e.g. $\varphi(20) = 8$. There are 8 numbers that are coprimes with 20 and are {1,3,7,9,11,13,17,19}.

- For instance, $\varphi(23) = 22$ because 23 is a prime number; ergo there are 22 counting numbers smaller than and mutually prime with 23. $\varphi(x) = x - 1$ equality is true such that x is a prime number.
- If x and y are mutually prime numbers (coprime), the Totient function has the feature of multiplication, and the following expression is true: $\varphi(xy) = \varphi(x) \times \varphi(y)$

Since the numbers p and q we generated are prime numbers, they are also mutually prime. By using the two examples given, we can write:

$$\varphi(p) = p - 1 \text{ and } \varphi(q) = q - 1$$

Since $n = p \times q$, due to the multiplication property of totient function: $\varphi(n) = \varphi(p) \times \varphi(q)$

$$\text{Then, } \varphi(n) = (p-1) \times (q-1)$$

4- The next operation requires the calculation and determining a random number e that satisfies the following terms: $1 < e < \varphi(n)$ and $\text{gcd}(e, \varphi(n)) = 1$ The number is a public data since it is used as the exponential value of the public key.

A random number in the $1 < e < \varphi(n)$ interval can be generated and used as the number e . The e number generated has to be mutually prime with $\varphi(n)$. For two integers to be coprime/mutually prime; their greatest common divisor should be 1; that is to say, $\text{gcd}(e, \varphi(n)) = 1$

Greatest Common Divisor (gcd)^{10 11 12}

The mathematician that discovered this algorithm is known as Euclid. Has been developed to find a solution to a construction problem. Say that you would like to have beautiful ceramics in your kitchen. You can calculate the length of one side of the tiles so that they will fit exactly to the length and width of your kitchen. Goodbye to half tiles! There is a nice animation on Wikipedia page that I would highly recommend you watch.

⁹ https://en.wikipedia.org/wiki/Euler%27s_totient_function

¹⁰ https://en.wikipedia.org/wiki/Euclidean_algorithm

¹¹ "The Euclidean Algorithm and Multiplicative Inverses"; <https://www.math.utah.edu/~fguevara/ACCESS2013/Euclid.pdf>

¹² Euclid Algorithm <https://www.youtube.com/watch?v=CG7ECG3e7vQ>

Let $1 < b < a$ be two positive integers. We can calculate the greatest number that divides both a and b . Let a be the dividend, b divisor, q quotient, and r remainder. Then, the division can be written as: $\frac{a}{b} = bq + r$
 Having b by itself on one side of the equation;

$$a = bq_1 + r_1$$

$$b = r_1q_2 + r_2$$

$$r_1 = r_2q_3 + r_3$$

The operations are repeated until the remainder becomes 0. $gcd(a, b)$ equals the last non-zero remainder. We simply repeat the process by dividing a by b and repeatedly dividing each remainder by the dividend until it reaches zero. Briefly shown as $gcd(a, b)$.

e.g. $gcd(8, 23) = ?$

$$23 = 8(2) + 7$$

$$8 = 7(1) + 1$$

$$7 = 1(7) + 0$$

$$gcd(8, 23) = 1$$

P.S. The numbers in parentheses are the multipliers.

If you do not wish to generate a random number, you can select such numbers as e that satisfies the $gcd(e, \varphi(n)) = 1$ term - i.e., numbers that are mutually prime with $\varphi(n)$. It would be efficient to select numbers with low Hamming weight (Hamming weight is the number of nonzero symbols in a series. For example, for 1101101110100, Hamming weight = 8), and with small values (eg $65537 = 00010000000000000001$, Hamming weight = 2). However, it has been observed that the selection of very small numbers (Ex: 3) creates security problems.

5- Number d is calculated using the equality $d \times e \equiv 1 \pmod{\varphi(n)}$.

We can not determine the number d . Since it has to satisfy the given equivalency, it needs to be calculated. What we need to do is to find the inverse of e due to $\pmod{\varphi(n)}$. We can read this expression as “the result of the operation $(d \times e)$ must have a mode value of 1 with base $\varphi(n)$ ”. A rather more understandable approach. Since we know the values of the numbers e and $\varphi(n)$, starting from 1, incrementing by 1 and adding it to d , we can calculate the result. Until the number d we test satisfies the equality, the cycle continues. The smaller the number e and the larger the number $\varphi(n)$, the more patient we must be - because this cycle can last forever. Moreover, since e and $\varphi(n)$ are mutually prime, unfortunately they do not have common divisors.

Fortunately, a modified version of the Euclidean algorithm in large numbers allows us to calculate the number d in a reasonable time. If e and $\varphi(n)$ are mutually prime, there is a modular inverse and it can be calculated.

Extended Euclidean Algorithm

We can use the Euclidean Algorithm to perform the multiplicative inversion in modular arithmetic of mutually prime numbers. When the Euclidean Algorithm is operated reversely, multiplicative inversion in modular arithmetic can be performed. Might be a bit complex - it is similar to driving a car in reverse gear - but not hard !

Let $0 < b < a$ be two positive integers. With a being the dividend, b divisor, q quotient, and r remainder: we wrote it in the Euclidean Algorithm as $a = bq + r$. This time, having the remainder r by itself on one side of the equation:

$$a = bq_1 + r_1 \quad \rightarrow \quad r_1 = a - bq_1$$

$$b = r_1q_2 + r_2 \quad \rightarrow \quad r_2 = b - r_1q_2$$

$$r_1 = r_2q_3 + r_3 \quad \rightarrow \quad r_3 = r_1 - r_2q_3$$

We obtain the equalities on the right side. This operation is carried out until the remainder is 0.

Suppose that gcd - gcd is the remainder of the equation before the remainder is 0. The algorithm must be run backwards from this equation. Let the step we obtained the - value be $r_3 = r_1 - r_2q_3$. If we replace the r_2 value in the equation with the previous $r_2 = b - r_1q_2$ equation, we get $r_3 = r_1 - (b - r_1q_2)q_3$. Thus, if we continue to put the remaining expressions from the end to the beginning and perform simplification operations, we reach the equation $r_n = ax + by$.

Especially, when $\text{gcd}(a, b) = 1$, and the numbers a and b are mutually prime, the equality $1 \equiv by \pmod a$ is satisfied. y is called the multiplicative inverse of $\pmod a$.

e.g. $1 = 8x + 23y$

Step 1: Calculate $\text{gcd}(8, 23)$

$$23 = 8(2) + 7 \rightarrow 7 = 23 - 8(2)$$

$$8 = 7(1) + 1 \rightarrow 1 = 8 - 7(1)$$

$$7 = 1(7) + 0 \rightarrow 0 = 7 - 1(7)$$

Step 2: run the algorithm backwards from the step gcd value had been found. By writing the factors in parentheses, not performing the multiplications and adding the factors; we aim to obtain the numbers 8 and 23.

$$1 = 8(1) - 7(1)$$

$$1 = 8(1) - (23 - 8(2))(1)$$

$$1 = 8(1) - 23 + 8(2)$$

$$1 = 8(3) - 23(1)$$

$$\Rightarrow x = 3, y = -1$$

The inverse of the number we want to find the modular inverse of would be the multiplier next to it. If the numbers in our example were $e = 8$ and $\varphi(n) = 23$, since we were looking for the inverse of e , by taking the multiplier near 8, we would have said that $d = 3$. In cases where the number d is negative, the following operation is done: $d = \varphi(n) + (-d)$.

e.g. If the multiplier was -1 , then $d = 23 + (-1)$, $d = 3$ would be found.

Whooh, take a deep breath.

B- MESSAGE ENCRYPTION AND DECRYPTION OPERATIONS

1- Message Encryption

Before encrypting, we need to know the (e, n) pair that is the public key of the person/institution we want to deliver the message to. In the next process, the data of the message we will send must be converted to an integer.

If the message we are going to send contains text, each letter of the text should be converted to the integer representation of a standard like ASCII or UTF (Unicode Transformation Format). For instance, the letter A corresponds to 65 in the ASCII table. If the message data consists of only numbers, you can encrypt it directly. Additional precautions are taken against plain text attacks by using the filling schemes before the encryption process is performed. In our example, we will not apply the fill process.

With c being the encrypted number, and $0 \leq m < n$: encryption is done by simply performing $c = m^e \pmod n$. The technical difficulties in this operation are large numbers and process of exponentiating. Even though the number m is small, the number e can consist of tens of digits. Multiplying a small number by itself millions of billions of times is a costly business. The number obtained as a result of exponentiating can be thousands of digits - expensive equipment is needed. Moreover, if you are going to do this for each unit integer of the message, even more expensive hardware is needed. Rather than the classical multiplication process, different methods are applied that accelerates the exponential operation.

Modular Exponentiation: Binary Exponentiation

Ask yourself: How to calculate $A^x \pmod C$ provided that x 's value is the power of 2 {1,2,4,8,16, 32, ...}? In a fast way!

We can write $A^2 \pmod C$ as $A^2 \pmod C = (A \pmod C \times A \pmod C) \pmod C$ by making use of the property of multiplication in modular arithmetic - $(A \times B) \pmod C = (A \pmod C \times B \pmod C) \pmod C$. The exponent of the number A is a power of 2.

e.g. Calculate $5^4 \pmod 11$

$$1-) 5^1 \pmod 11 = 5$$

$$2-) 5^2 \pmod 11 = (5^1 \pmod 11 \times 5^1 \pmod 11) \pmod 11$$

3-) By using $5^1 \pmod 11 = 5$ we found earlier,

$$5^2 \pmod 11 = (5 \times 5) \pmod 11 = 3$$

$$4-) 5^4 \pmod 11 = (5^2 \pmod 11 \times 5^2 \pmod 11) \pmod 11$$

5-) By using $5^2 \bmod 11$ we found earlier,

$$5^4 \bmod 11 = (3 \times 3) \bmod 11 = 9$$

6-) Proving, $5^4 = 625 = 625 \bmod 11 = 9$

This method seems pretty simple and fast. So, how do we calculate $A^x \bmod C$ without x 's value being the power of 2? Our work got a little longer, but we're not desperate here. What we are going to do is to write x in binary.

e.g. Calculate $5^{23} \bmod 11$

Writing the exponential in binary: $23 = 00010111 = (2^0 + 2^1 + 2^2 + 2^4)$

$$5^{23} \bmod 11 = 5^{(1+2+4+16)} \bmod 11 = (5^0 \times 5^1 \times 5^2 \times 5^4) \bmod 11$$

We have calculated value in the previous example. Moving on:

1-) $5^8 \bmod 11 = (5^4 \bmod 11 \times 5^4 \bmod 11) \bmod 11$

2-) Using $5^4 \bmod 11$ we found earlier:

$$5^8 \bmod 11 = (9 \times 9) \bmod 11 = 4$$

3-) $5^{16} \bmod 11 = (5^8 \bmod 11 \times 5^8 \bmod 11) \bmod 11$

4-) Using $5^8 \bmod 11$ we found earlier:

$$5^{16} \bmod 11 = (4 \times 4) \bmod 11 = 5$$

5-) Substituting:

$$5^{23} \bmod 11 = 5^{(1+2+4+16)} \bmod 11 = (5 \times 3 \times 9 \times 5) \bmod 11$$

$$5^{23} \bmod 11 = 4$$

2- Message Decrypting

Decryption process is just like the encryption process. The only difference is that we have to use the private key pair (d, n) . With m being the decrypted number, c encrypted number and $0 \leq c < n$, the decryption of the message is done by performing the $m = c^d \pmod n$ operation. This is done for all integer numbers of the encrypted message. In the next operation, if the data of the incoming message contains a text, each decoded number should be translated from the table of a standard such as ASCII or UTF (Unicode Transformation Format) or the received data can be saved to a file and processed by the program that recognizes the file format.

Sample encryption and decryption operation:

To make the example simple and understandable, we are going to work with small numbers.

1. Let $p = 47, q = 83$ be prime numbers
2. By using the equality $n = p \times q$, the number n is calculated $n = 47 \times 83 = 3901$.
3. By using the equality $\varphi(n) = (p-1) \times (q-1)$, $\varphi(n) = (47-1) \times (83-1) = 3772$
4. Numbers $\varphi(n)$ and e should be mutually prime. Therefore, let $e = 5$. No need for gcd test.
5. Number d is calculated using the equality $d \times e \equiv 1 \pmod{\varphi(n)}$. $d \times 5 \equiv 1 \pmod{3772} \quad d = ?$

First, let's calculate gcd using the Euclidean algorithm

$$\begin{aligned}
 1 &= 5x + 3772y \\
 3772 &= 5(754) + 2 \rightarrow 2 = 3772 - 5(754) \\
 5 &= 2(2) + 1 \rightarrow 1 = 5 - 2(2) \\
 2 &= 1(2) + 0 \rightarrow 0 = 2 - 1(2) \\
 \\
 1 &= 5 - 2(2) \\
 1 &= 5 - (3772 - 5(754))(2) \\
 1 &= 5 - (3772 - 5(1508)) \\
 1 &= 5(1509) - 3772(1)
 \end{aligned}$$

Here, we are concerned with p : that is to say, the multiplier near 5. Since it is positive, we are going to use it directly, $d = 1509$

As the result of our calculations, we found the public $(e, n) = (5, 3901)$ and private keys $(d, n) = (1509, 3901)$. Let's try it now. Citing the day the article is written, let the text to be encrypted be "ÇANAKKALE GEÇİLMEZ".

1- In order to be able to encrypt it using the RSA algorithm, we convert it to an integer string using the ASCII table.

195 is written instead of Ç. We process each letter of the text and finish our preparation. The corresponding ASCII string of integers of ÇANAKKALE GEÇİLMEZ is 195, 135, 065, 078, 065, 075, 075, 065, 076, 069, 032, 071, 069, 195, 135, 196, 176, 076, 077, 069, 090 .

2- By using the public key $(e, n) = (5, 3901)$, $c = m^e \pmod{N}$ operation is applied to each integer. Taking Ç, with value being 195 for example:

$$c = 195^5 \pmod{3901} = 281950621875 \pmod{3901} = 3177, c = 3177$$

The letter "Ç", encrypted with the public key, corresponds to 3177. We apply this operation to the other numbers in the string.

After the operation is done, we got the following string: {3177, 1201, 0491, 1357, 0591, 2258, 2258, 0591, 0208, 3419, 1931, 1247, 3419, 3177, 1201, 2829, 1396, 0208, 1188, 3419, 3112}. This string is the encrypted version of our text.

3- By using the private key $(d, n) = (1509, 3901)$, we apply the $m = c^d \pmod{N}$ operation to each integer. The first integer we are going to decrypt is 3177.

$$m = 3177^{1509} \pmod{3901} = 34 \dots \pmod{3901} = 195$$

The result of the 3177^{1509} operation is a number with 5285 digits. For those who would want to see this number can use an online large number calculator¹³. Just write the $3177^{1509} \% 3901$ formula in the expression field and observe. The 195 number we obtained is the ASCII correspondent of the letter Ç. Try the other elements of the encrypted integer string to challenge yourself: {3177, 1201, 0491, 1357, 0591, 2258, 2258, 0591, 0208, 3419, 1931, 1247, 3419, 3177, 1201, 2829, 1396, 0208, 1188, 3419, 3112}.

Epilogue

You might be wondering how prime factoring large numbers protects the RSA algorithm or how is this operation also RSA's weakness. Now that you know what RSA is, remember how we calculated the $n = p \times q$ and $\phi(n) = (p-1) \times (q-1)$ equalities and the number n and $\phi(n)$. After all, (e, n) public keys are numbers known by everyone. What ensures confidentiality and decrypts the encrypted message is the number d calculated using the equality $d \times e \equiv 1 \pmod{\phi(n)}$. If the attacker successfully finds the numbers p, q in the equality $n = p \times q$ by prime factoring, it would be very easy for them to calculate the number $\phi(n) = (p-1) \times (q-1)$, then the inverse of the number e relative to $\pmod{\phi(n)}$ and calculate the number d . The rest is history. We wouldn't want to let them do this, right?

https://www.arkakapidergi.com - Arka Kapı Dergi 7.Sayı RSA Uygulaması

Yeni Örnek

Anahtar Boyu (bit) = 32 40 48 56 64

$p =$ 2117420443

$q =$ 1817865251

$n = p \times q =$ 3849185045086726193

$\phi(n) = (p - 1) \times (q - 1) =$ 3849185041151440500

E Sayısı Üret $e =$ 1817865251

$d =$ 2613704185817644751

Şifrele **Çöz**

Anahtar Boyu (bit) =

Bir bahar akşamı rastladım size
Sevinçli bir telaş içindeydiniz
Derinden bakınca gözleriniz
Neden başınızı öne eğdiniz

İçimde uyanan eski bir arzu
Dedi ki yıllardır aradığın bu

0284E6CDBDF225512C42BD61C328501301064AC9BCE4
51702EBE3C8337B3B04E2F55D0D3DAB0031618BA2771
270B55D625611324D5A5C46018BA2771270B55D601064
AC9BCE451702EBE3C8337B3B04E18BA2771270B55D62
7A39F57C6E9B854277717FA1104C2FF18BA2771270B55
D60AED19CD35215B38012D0E1CDB59AEE42EBE3C833
7B3B04E01064AC9BCE4517018BA2771270B55D62DCA8

Arka Kapı Dergisi 7.sayı
için Kriptoloji eğitimi amaçlı
hazırlanmıştır.

ARKAKAPI
SİBER GÜVENLİK DERGİSİ

Bir bahar akşamı rastladım size
Sevinçli bir telaş içindeydiniz
Derinden bakınca gözleriniz
Neden başınızı öne eğdiniz

İçimde uyanan eski bir arzu
Dedi ki yıllardır aradığın bu

The sample application is written in VS2012 C# language. Made use of the sources over the net pretty much. No need to reinvent the wheel. Yet, we have the responsibility to understand what exists. This application is written in a non-professional manner, for educational purposes. There are ready-to-use RSA and cryptology classes in the .NET

architecture. Using these classes directly, you can write an encryption application with more performance and a few lines of code. Within the application, the random number generator and some other procedures are limited to a 64-bit integer. You can remove this limitation yourself, adapt the data types to the Biginteger class and encrypt with higher (1024 bit) digits. However, in order to operate with higher digit keys, it is recommended to replace the prime number test with methods of higher performance.

I would like to thank dear Mr. Halit İnce for his precious suggestions and contributions in the preparation of the article.

Link to sample application: <https://bit.ly/2UiRGLR>



Prophecies for the Next 30 Years of cybersecurity

Predicting the future plays a crucial role in cybersecurity as in any sector. Knowing where the cybersecurity sector will move in the future means profit for companies and strategic superiority for states. Consulting firms that make such estimates exist in the world. However, their future forecasts are short-term like 2-5 years. In this article, we will imagine where we can end up within the next 30-40 years.

It may seem impossible to predict such a distant future in the world of technology, which gets completely different every five years. However, in the cybersecurity world, it is possible to see a concentrated projection of the events of humanity in the historical process.

In ancient times without security cameras and forensics, it was very difficult to identify criminals. By the 1800s, forensics had advanced and being a detective became a common profession. Today, the detection of criminals has become much easier. According to this projection, the cybersecurity world in 2018 is similar to the physical world of the 1800s. If we examine the historical process after the 1800s, we can predict the changes that cybersecurity will experience in the next 30-40 years with the help of some imagination.

In this article, I am going to mention three main events that I anticipate will take place around 2030, 2040 and 2050.

2030 - The Unemployment Problem in the Security Sector

At the beginning of the 2000s, while the web and in-

ternet technologies were becoming widespread in the world, the security side of the business did not disturb people's minds. Even though the hacker theme was used in the adventure films that emerged in this period, the damages they could cause to our daily lives did not go beyond a science fiction scenario. But as the years progressed, technology has penetrated every corner of everyday life. Therefore, the security part of the business became a critical situation. However, there is still a big shortage in the number of required security personnel. Therefore, the institutions that make predictions for the future predict that the need for security personnel will increase as digitalization will increase in the future.

This reminds me of the history of elevator operators. When the elevator was first invented, the help of a human operator was required to operate it. A person who considered that elevators would become widespread all over the world at that time could say that the need for human operators would increase. But with the arrival of electronic button systems to the elevators, the need for humanity ended and a profession was thus lost.

There have been great changes in the security sector since the 2000s. Thanks to the importance of security, defensive security technologies have progressed. Companies and governments have started to invest their assets heavily in the security sector. Researchers have begun to detect and close security vulnerabilities in open source software. Therefore, when we compare 2008 and 2018, we can say that security on a global scale has considerably improved. So where is it going from now on?

I listed the phases of the security software as follows:

- 0) The initial phase (... - 2001)
- 1) Ease of use phase (2001-2009)
- 2) Maturity phase (2009-...)
- 3) Artificial intelligence phase (...-...)

We do not exactly know when we will move from the current maturity stage to the artificial intelligence stage. But we can observe that the course is definitely in that direction with many different indications. For example, defensive security products such as IDS and SIEM are about to reach the point of working without human need. On the offensive side, although they are still in their infancy, we can see that artificial intelligence hacker software has emerged. Although these programs are experimental and expensive, they will be more stable and cheaper in the future. Therefore, this software will be more cost-effective and more efficient than a human employee.

Considering the trend, I think that we will pass to this period by 2030. So, what happens when this period comes? A large part of both defensive and offensive security will become automated. The need for a human being may be limited to correction from an overlooked eye and an emerged problem. In the cyber warfare part, the human need of states will probably continue but the private sector will prefer to get mechanized in this field. By 2030, middle and lower level security specialists who have not developed themselves in different disciplines such as programming, artificial intelligence, and its sub-branches will face unemployment.

2040 - Virtual Passport and Virtual Visa

It is a well-known fact that companies like Facebook and Google record the behaviour of individuals by cookies or other methods. Although this personal data is only used informally by these companies or the US government, I think it will be different in the future. Because one of the most important reasons why cybercrime cannot be stopped nowadays is that the real identity of the criminals cannot be determined. While individual profiles of companies like Google can sometimes help identify criminals, they don't provide a defi-

nite solution. In addition, services such as TOR help individuals to hide their actual IP address.

The fact that TOR and other VPN services still work is the result of the Internet being open to everyone. For example, a person living in Indonesia may connect to "turkiye.gov.tr" and may cause damage here if he/she finds a vulnerability. "turkiye.gov.tr" site admins can think that anyone living in Indonesia does not need to enter this site, or can even block the entire world except Turkey. But in this case, how will Turkish citizens living abroad enter the site? Country blocking cannot be a permanent solution to such issues. The permanent solution is the virtual passport system, which I think will be available in the future.

Imagine a company called "Evilcorp", which aims to stop cybercrimes by preventing people from surfing the internet anonymously. People can get an electronic passport from Evilcorp with a real identity. After the person receives an electronic passport, every request he/she sends on the Internet will include this passport information. The websites integrated with Evilcorp will be able to check virtual passports for visitors. For example, the website may block a country's citizens completely, but visa may be granted to students and academicians of that country. The big cloud companies of that era will be integrated with Evilcorp, and most of the internet will not be able to roam without virtual passports.

2050 - End of Individual Hacking and the Age of Hackers

Throughout history, humanity has been exposed to different criminal trends and groups and has developed methods of tackling them. An example of this is sea piracy. The history of piracy, which began in the 14th century B.C., lasted until the 1800s. Hackers are likened to pirates in various cultures. For example, the word "hacker" means "computer pirate" in the Turkish Language Association. This analogy is not unreasonable. We can compare the Internet to the ocean, computers to ships and hackers to pirates. So how did the centuries-old piracy tradition end in the 1800s? From what I've read, there are two things that are effective here: first, the British navy's patrols had increased in number, and their patrol was more than pirates. The second was the closure of the ports where the pirates had traded. Pirates were forced to abandon this ancient tradition of crime with the interruption of income and the increas-

ing pressure. The so-called endless piracy has become a thing of the past.

Of course, the only motivation of hackers is not financial gain. Therefore, such hacking activities will not end even if the earning path is cut just like pirates. However, there are two main issues that will end hacking in the future. First, with the virtual passports mentioned in the previous section, hacking will become a crime that requires great courage. Because of that, every internet user will be recorded in each step. The more difficult it is to kill a person today and not get caught, the more hacking will be in the future. Besides, when we take a look at the 2000s to the present, we see that defensive security technologies have gained a lot of power. This

will probably continue to happen, and hacking a system in the future will become something that only very big actors can achieve.

As a result of all, hacking activities will be limited to military and intelligence methods and individual hacking will become history. The hacking culture will probably not end in such a short period of time. However, it does not seem possible to continue such activities as a culture for a long time if these do not go down to individuals. 200 years from now, historians will describe hackers as people existed between 1980 and 1950, and young people of that time will join the costume parties with black hoodies with hacker emblems.



CALL FOR PAPERS

АРКАКАПИ

Do you want your article to be published on Arka Kapi Magazine? Submit now to be featured in the next issue! Your article can be of any title as long as it fits to the cyber security context. Make sure it's an original article that isn't previously published elsewhere.

Email your articles to:
editor@arkakapimag.com

FEEDBACK

Got any feedback about Arka Kapi Magazine? Found a bug? Want us to add or remove something? Let us know!

follow us

Don't miss the news!



arkakapimag