Privacy and security risks

RISKS

Three main risks from emails:

1. **Email spamming**: unsolicited electronic messages
2. **Email tracking**: emails can be used to track user actions
3. **Email phishing**: social engineering attacks based on a fraudulent ("spoofed") message

SPAM: origin of the term

A brand of canned cooked pork introduced by Hormel in 1937.
It gained popularity worldwide after its use during World War II.

It has become the subject of a number of appearances in pop culture, notably a Monty Python sketch, **which repeated the name many times**, leading to its name being borrowed for unsolicited electronic messages.

UNWANTED E-MAIL MESSAGES

- SPAM = unwanted ads
  - both normal and low quality merchandize (drugs, pharmacy, dating, online sex, pirated software/multimedia etc.)
- frauds/malware
  - "write here your username/password"
  - "write here your credit card number"
  - "help me to retrieve $ 20 000 000 ..."
  - "you haven't claimed your € 500 prize"
  - loans and funds at lowest rates
  - "I'm so lonely and looking for love..."
  - "you won the lottery"
  - "the message you have sent is undeliverable"
  - “invoice to be paid: click here”
- e-mail chain letters
  - exponential growth
- all of above, joint to low-quality automatic language translation

we'll use the generic terms spam or junk for denoting unwanted or undesirable e-mail messages

GOALS OF SPAM

- sell products/services (aggressive marketing)
- sell low-qualities/fake/expired goods/medicines (low prices)
- distribute/spread malware (viruses, worms, Trojan horses, backdoors, rootkits etc.) and grayware (adware, spyware, dialers etc.)
    - computer can be enrolled/controlled for participation in (future) attacks
    - Internet activity (browsing, instant messaging and other social activity) can be monitored, users can be profiled
    - audio/video sessions can be recorded
    - collect (any) data on you and on your contacts (databases are built to the purpose of digital identity thefts)
- phishing
    - username/password stealing, credit card data capture, frauds etc.
    - often based on malicious links
- validate e-mail addresses
    - can be re-sold at a higher price
    - based on HTML images and links

COMMON SENSE

- disable HTML messages or, at least, disable download of remote images
  - prevent the sender to validate our e-mail address
- don't click links (specially if tiny or IP-based URLs)
  - could redirect to bad web sites containing malware/spyware
- don't open unknown/unexpected attachments
  - they may contain malware/spyware
  - executables (.exe, .app, .bat etc.), documents(.doc, .pdf etc.) and others (.src, ...)
- use anti-spam filter
- don't participate with chain letters: google their contents!
- protect and respect privacy of other recipients
  - be careful in e-mail forwarding (don't uselessly disclose e-mail addresses)
- don't click "delete me"
  - may validate your email address
  - OK with known senders

EMAIL TRACKING

By reading an email, you may reveal sensitive information:

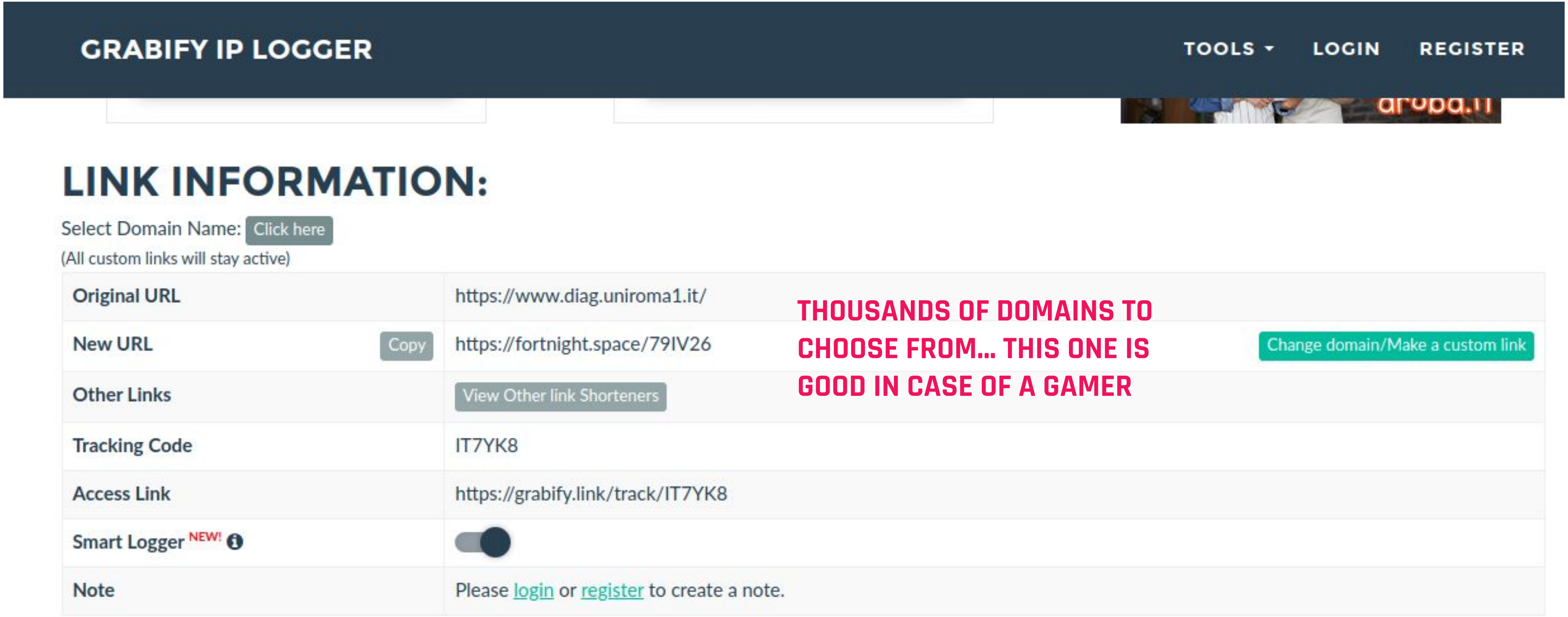
- whether the email was read: the email address is valid and the user likely read the content.
- the IP address of the victim, which may be used to perform direct attacks or know the approximate location (country and city)
- other info (browser, device, etc.) about the user sent by the browser when performing a request

EMAIL TRACKING (2)

This can be easily be achieved by embedding:

- external images: a unique URL is associated to each email message and the attacker only needs to check the *access log* on its server.
- a shortened URL: if the user opens the URL, a page tracks its info and immediately redirect him to a valid page
- an “unsubscribe” URL: expert users may trick into this one…

EXAMPLE: ONE TRACKING SERVICE



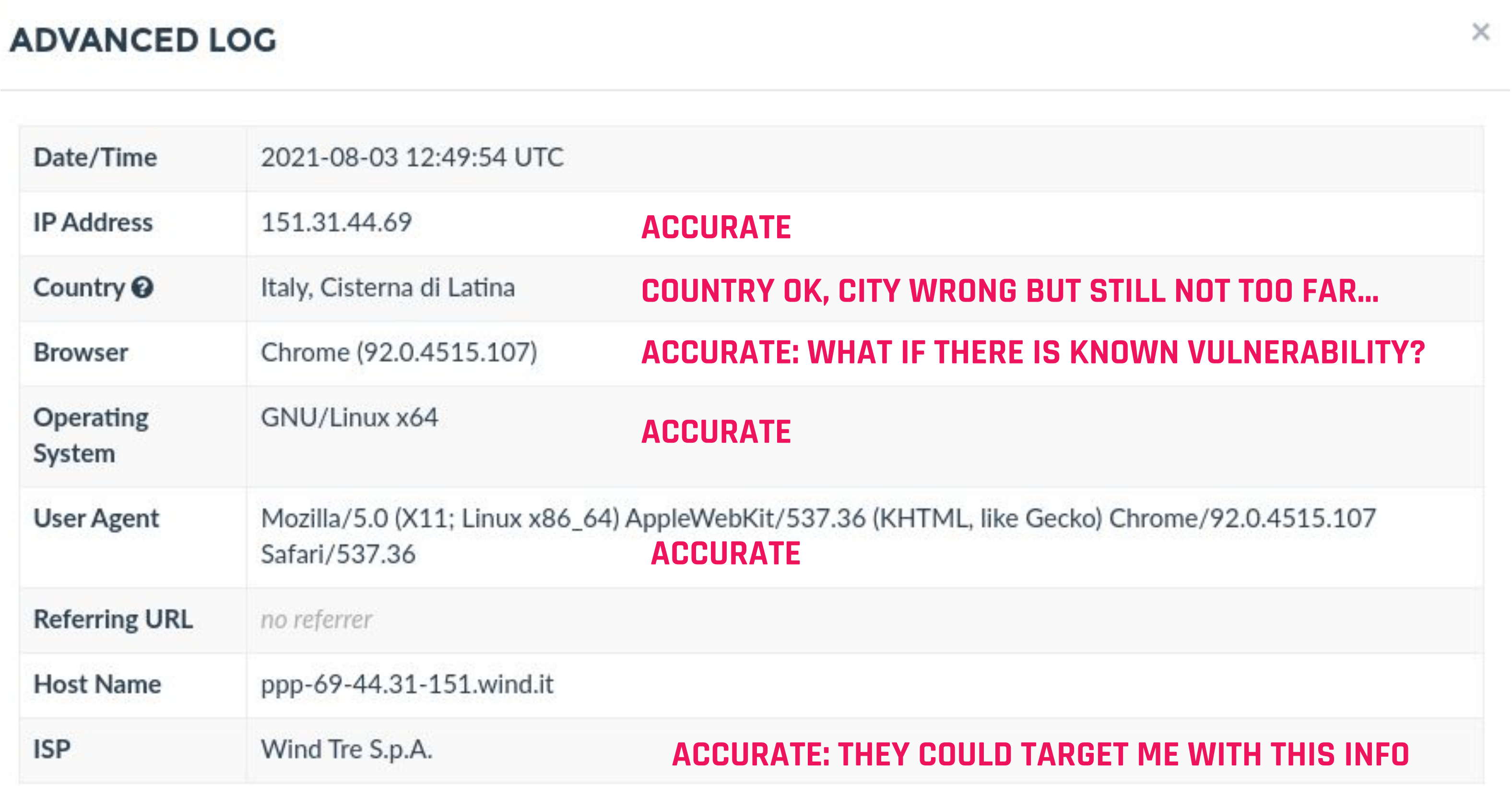
LINK INFORMATION:

Select Domain Name: Click here
(All custom links will stay active)

Original URL	https://www.diag.uniroma1.it/
New URL Copy	https://fortnight.space/79IV26 Change domain/Make a custom link
Other Links	View Other link Shorteners
Tracking Code	IT7YK8
Access Link	https://grabify.link/track/IT7YK8
Smart Logger NEW!	●
Note	Please login or register to create a note.

THOUSANDS OF DOMAINS TO CHOOSE FROM… THIS ONE IS GOOD IN CASE OF A GAMER

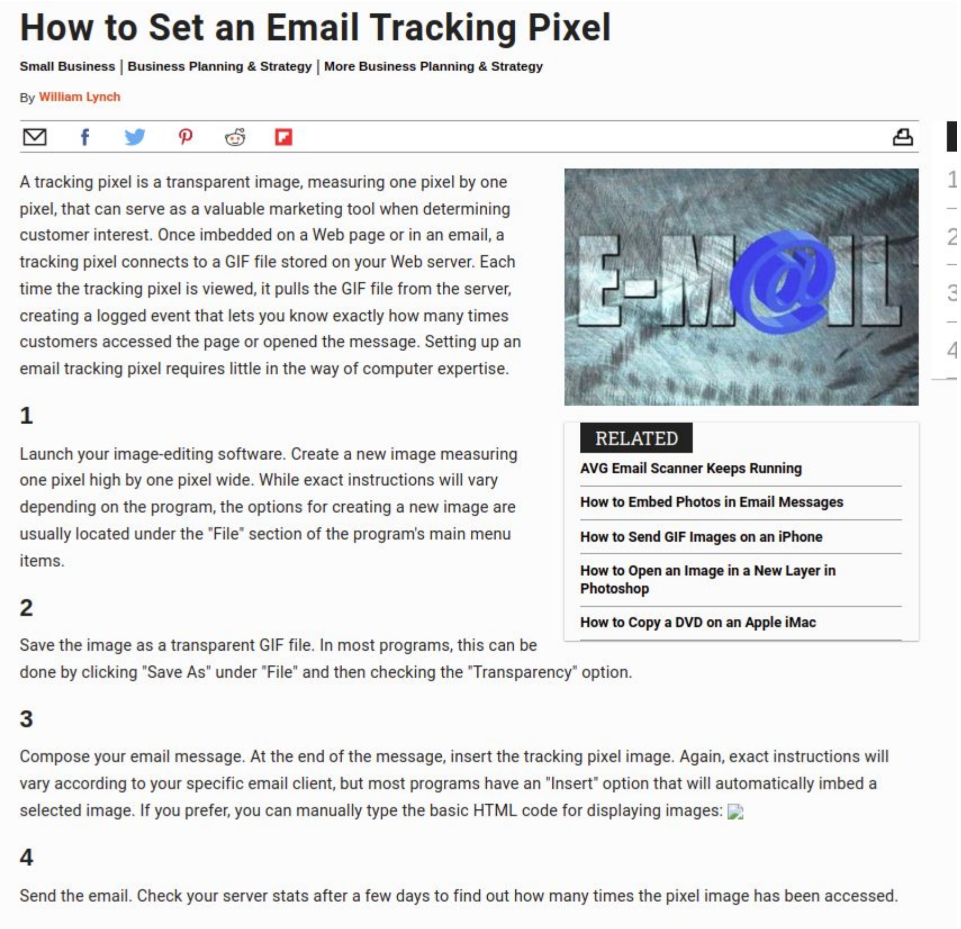
AFTER CLICKING THE URL…



ADVANCED LOG

Date/Time	2021-08-03 12:49:54 UTC	
IP Address	151.31.44.69	ACCURATE
Country ❓	Italy, Cisterna di Latina	COUNTRY OK, CITY WRONG BUT STILL NOT TOO FAR…
Browser	Chrome (92.0.4515.107)	ACCURATE: WHAT IF THERE IS KNOWN VULNERABILITY?
Operating System	GNU/Linux x64	ACCURATE
User Agent	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/92.0.4515.107 Safari/537.36	ACCURATE
Referring URL	*no referrer*	
Host Name	ppp-69-44.31-151.wind.it	
ISP	Wind Tre S.p.A.	ACCURATE: THEY COULD TARGET ME WITH THIS INFO

DIY TRACKING SYSTEM USING AN IMAGE…



How to Set an Email Tracking Pixel

Small Business | Business Planning & Strategy | More Business Planning & Strategy

By William Lynch

A tracking pixel is a transparent image, measuring one pixel by one pixel, that can serve as a valuable marketing tool when determining customer interest. Once imbedded on a Web page or in an email, a tracking pixel connects to a GIF file stored on your Web server. Each time the tracking pixel is viewed, it pulls the GIF file from the server, creating a logged event that lets you know exactly how many times customers accessed the page or opened the message. Setting up an email tracking pixel requires little in the way of computer expertise.

1

Launch your image-editing software. Create a new image measuring one pixel high by one pixel wide. While exact instructions will vary depending on the program, the options for creating a new image are usually located under the "File" section of the program's main menu items.

2

Save the image as a transparent GIF file. In most programs, this can be done by clicking "Save As" under "File" and then checking the "Transparency" option.

3

Compose your email message. At the end of the message, insert the tracking pixel image. Again, exact instructions will vary according to your specific email client, but most programs have an "Insert" option that will automatically imbed a selected image. If you prefer, you can manually type the basic HTML code for displaying images:

4

Send the email. Check your server stats after a few days to find out how many times the pixel image has been accessed.

RELATED

- AVG Email Scanner Keeps Running
- How to Embed Photos in Email Messages
- How to Send GIF Images on an iPhone
- How to Open an Image in a New Layer in Photoshop
- How to Copy a DVD on an Apple iMac

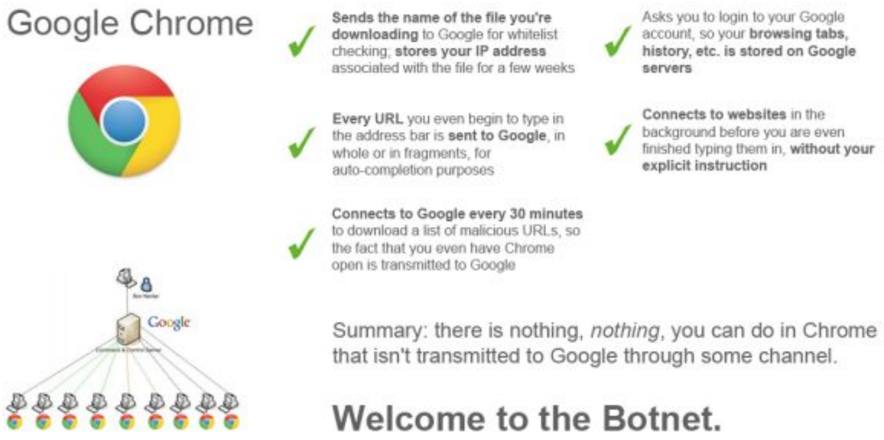
ALL WEB COMPANIES ARE TRACKING US...

Look for URLs to third-party website found on:

- Google
- Facebook
- Twitter
- Most mobile apps

Also, some software (e.g., browser) also tracks user actions. They do this for several reasons:

- Security: they can always blocks malicious URL
- Profiling: they know everything you do...



Google Chrome

✓ **Sends the name of the file you're downloading** to Google for whitelist checking; **stores your IP address** associated with the file for a few weeks

✓ Asks you to login to your Google account, so your **browsing tabs, history, etc. is stored on Google servers**

✓ **Every URL** you even begin to type in the address bar is **sent to Google**, in whole or in fragments, for auto-completion purposes

✓ **Connects to websites** in the background before you are even finished typing them in, **without your explicit instruction**

✓ **Connects to Google every 30 minutes** to download a list of malicious URLs, so the fact that you even have Chrome open is transmitted to Google

Summary: there is nothing, *nothing*, you can do in Chrome that isn't transmitted to Google through some channel.

Welcome to the Botnet.

PHISHING AND SPEAR PHISHING

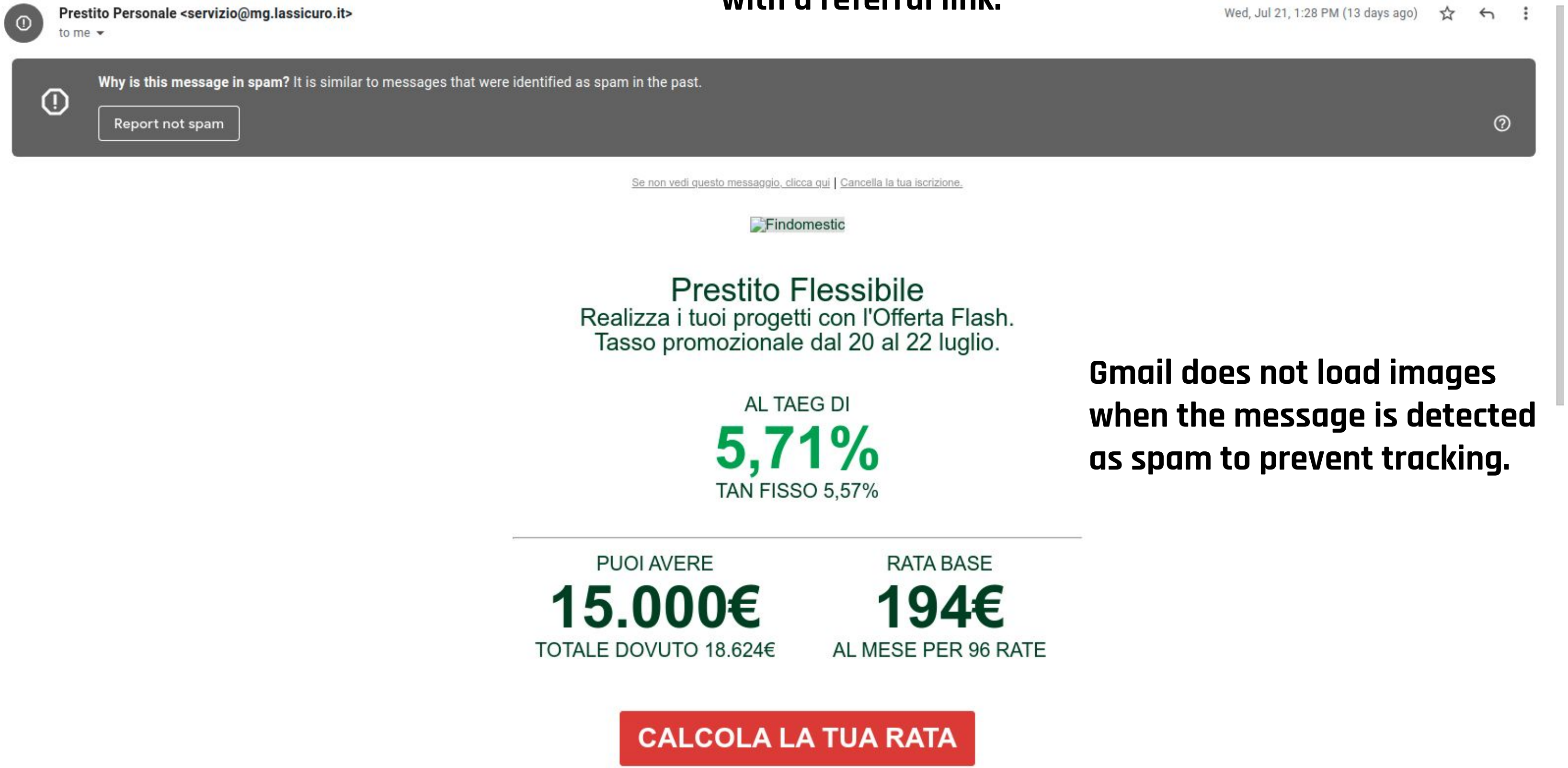
A social engineering technique where an attacker sends a fraudulent ("spoofed") message designed to trick a human victim into revealing sensitive information to the attacker or to deploy malicious software on the victim's infrastructure like ransomware.

Two types:

- **“general” phishing**: mass campaign targeting millions of users. Typically, the user can detect them due to typos and inconsistencies. Relatively easy to detect for email providers that can analyze millions of email boxes.
- **spear phishing**: targeted attack to a category of users or even a specific single user. Done by a motivated attacker. Hard to automatically detect. **Hard to detect for 99% of the users…. me and likely you are in this 99%!**

EXAMPLE OF SPAM

This is spam because when you open the link, they are actually redirecting to the right website with a referral link.



Prestito Personale <servizio@mg.lassicuro.it>
to me

Wed, Jul 21, 1:28 PM (13 days ago)

Why is this message in spam? It is similar to messages that were identified as spam in the past.

Se non vedi questo messaggio, clicca qui | Cancella la tua iscrizione.

Findomestic

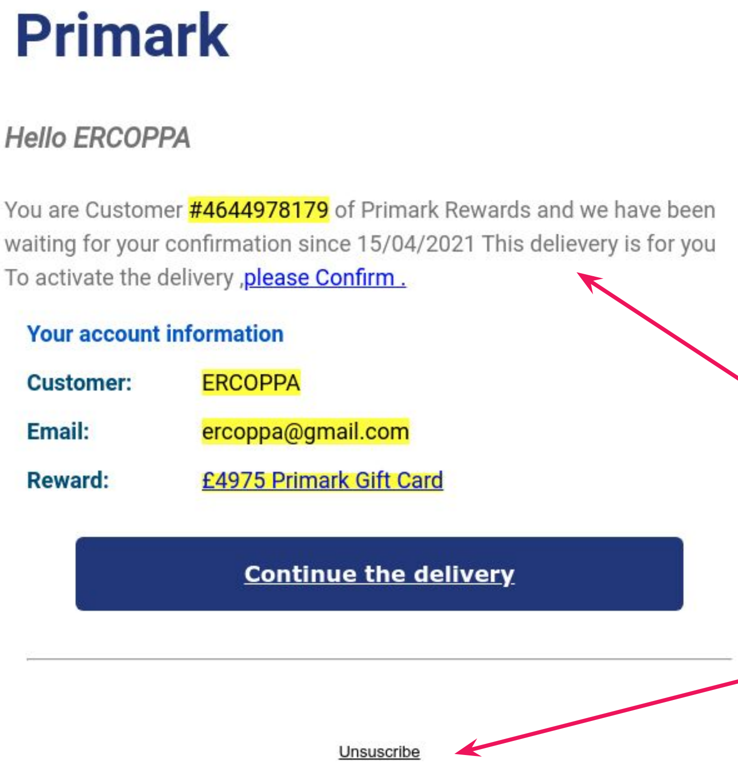
Prestito Flessibile
Realizza i tuoi progetti con l'Offerta Flash.
Tasso promozionale dal 20 al 22 luglio.

AL TAEG DI
5,71%
TAN FISSO 5,57%

PUOI AVERE	RATA BASE
15.000€	194€
TOTALE DOVUTO 18.624€	AL MESE PER 96 RATE

Gmail does not load images when the message is detected as spam to prevent tracking.

EXAMPLE OF PHISHING



Primark

Hello ERCOPPA

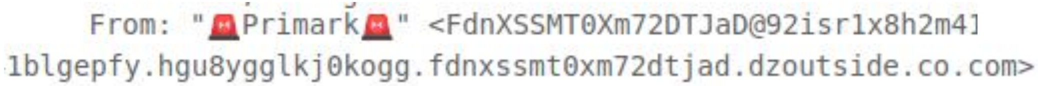
You are Customer **#4644978179** of Primark Rewards and we have been waiting for your confirmation since 15/04/2021 This delievery is for you To activate the delivery ,please Confirm .

Your account information

Customer:	ERCOPPA
Email:	ercoppa@gmail.com
Reward:	£4975 Primark Gift Card

This is phishing because they are impersonating a brand.

The name shown is faking the brand:

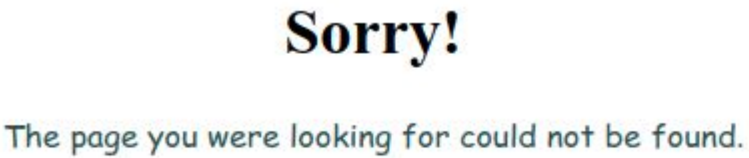


```
From: "🚨Primark🚨" <FdnXSSMT0Xm72DTJaD@92isr1x8h2m4l
1blgepfy.hgu8ygglkj0kogg.fdnxssmt0xm72dtjad.dzoutside.co.com>
```

The address is not from Primark but it so long on purpose. Why?

Typo. No real-world big brand will likely mistype their messages.

Bad URLS: if you open them, they show:



Sorry!

The page you were looking for could not be found.

EXAMPLE OF SPEAR PHISHING (CENSORED)

Subject: Costo netto e lordo Contratto XYZ XYZ per aggiornamento budget XYZ 2019

Password archivio: 6209

<Head of the company>
<Institutional email address of the head of the company>
--
<long (real!) email thread with multiple users talking about the contract>

<ZIP ATTACHMENT WITH PASSWORD> // it contained a doc file with a malicious macro

HOW TO (TRY TO) SURVIVE?

- **SPAM**: use a spam filter; most web client have one
- **TRACKING**: do not open links; use anti-tracking features (e.g., Gmail does not show images if the message is marked as spam). Still, “good services” will track you… no matter what you do.
- **GENERAL PHISHING**: pay attention to the content of the message; use a spam filter.
- **SPEAR PHISHING**: use your brain; keep in mind that **a motivated attacker will find a way to trick you**. Keep your software up-to-date!

Email Validation Systems

E-MAIL VALIDATION SYSTEMS

- **Sender Policy Framework (SPF)**
  - prevents e-mail spam by detecting email spoofing through verification of sender IP addresses
  - RFC 4408

- **DomainKeys Identified Mail (DKIM)**
  - allows to check that incoming mail from a domain is authorized by that domain's administrators and that the email (possibly including attachments) has not been modified during transport
  - RFC 4871

- **Domain-based Message Authentication, Reporting and Conformance (DMARC)**
  - Extends SPF and DKIM with different policies (e.g., how to report spam from a domain?)
  - RFC 7489

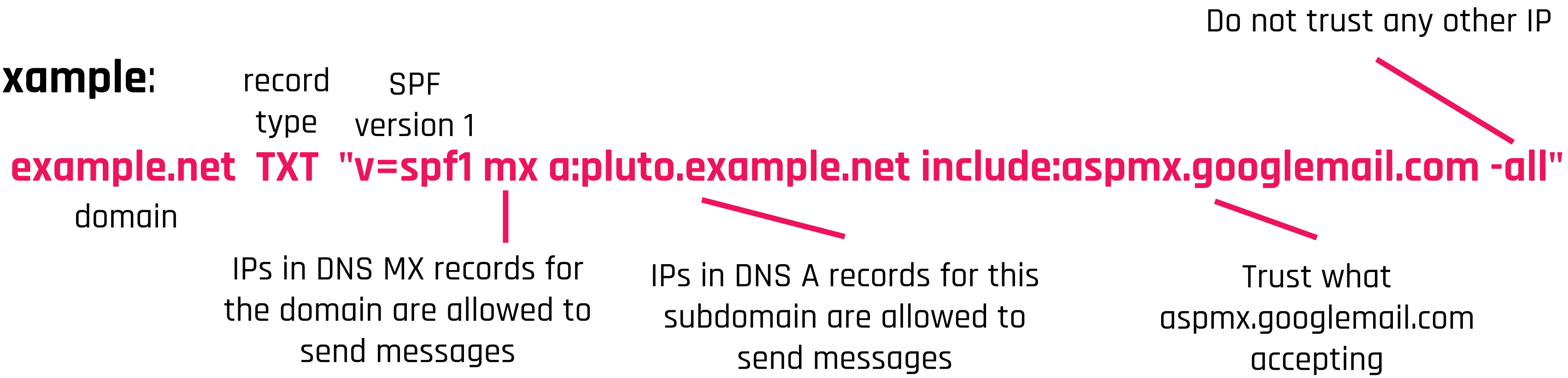
- **Authenticated Received Chain (ARC)**
  - A message may traverse a chain of SMTP server: ARC validates the entire chain, even when the message could have been modified (for good reasons).
  - RFC 8617

Sender Policy Framework (SPF)

IDEA:

- a domain publish a DNS TXT record containing the IPs allowed to send messages
- an SMTP server checks the TXT record to validate the sender's IP
- The sender's IP is taken by looking at **Return-Path (MAIL FROM)**

Example:



example.net TXT "v=spf1 mx a:pluto.example.net include:aspmx.googlemail.com -all"

domain

record type

SPF version 1

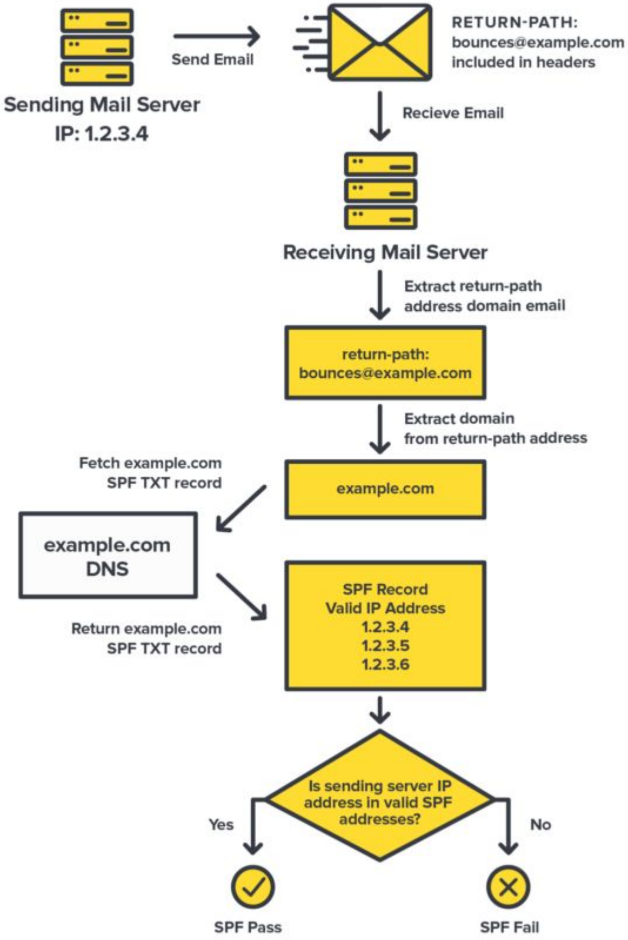
IPs in DNS MX records for the domain are allowed to send messages

IPs in DNS A records for this subdomain are allowed to send messages

Trust what aspmx.googlemail.com accepting

Do not trust any other IP

Reference: Wikipedia SPF



[image credits]

SPF: PROBLEMS

- SPF only validates the **Return-Path** but does nothing for **From,** which is the most frequently spoofed field

- SPF breaks when a message is forwarded (true only for some forwarding methods): the Sender's IP is not the expected one. However, there could be good reasons to forward emails (e.g., mailing list).

- Just because a message fails SPF, doesn't mean it be will *always* be blocked from the inbox – it's one of several factors email providers take into account.

- **SPF does not authenticate the mail content: what if the content has been altered?**

DomainKeys Identified Mail (DKIM)

IDEA:

- the (server of the) sender signs the message using his private key
- this requires to add a DKIM header in the message
- different parts of the message could be signed: **From** is mandatory
- the domain publish a DNS TXT record containing info about the public key. Selectors are used to define different keys for different purposes/subdomains
- the receiver validate the message using the public key

Reference: Wikipedia DKIM; Example of signing/verifying DKIM signatures: dkimpy

DKIM: example

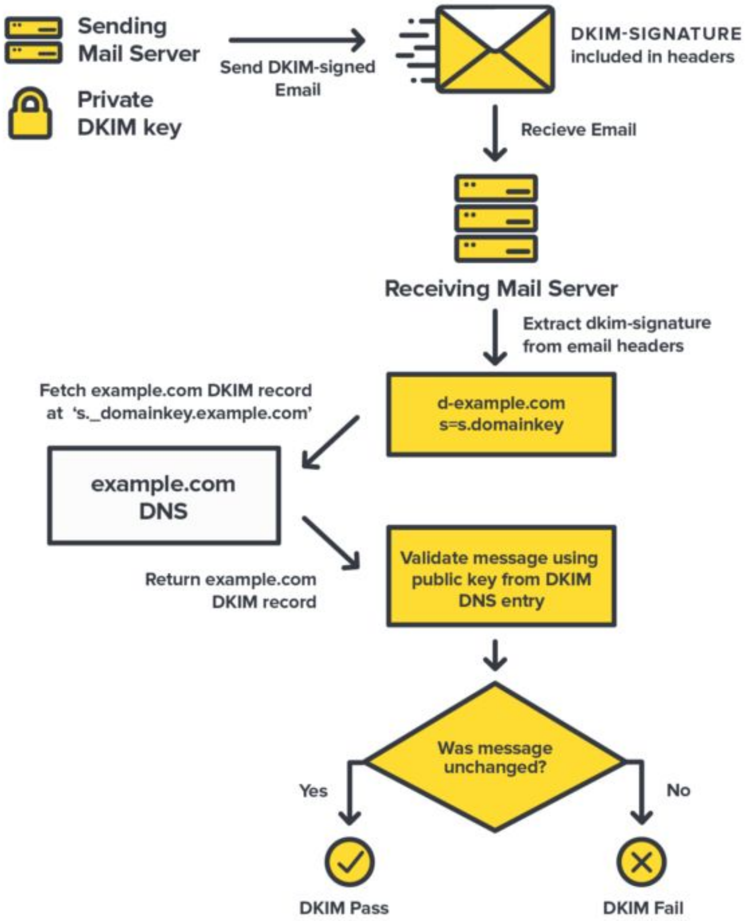
TXT record:

brisbane._domainkey.example.net TXT "v=DKIM1; k=rsa; p=<base64-pubkey>"
selector   fixed   domain

Message Header:

DKIM-Signature: v=1; **a=rsa-sha256**; d=example.net; s=brisbane;
c=relaxed/simple; **q=dns/txt**; t=1117574938; x=1118006938; timestamp, expire time
h=from:to:subject:date:keywords;
bh=MTIzNDU2Nzg5MDEyMzQ1Njc4OTAxMjM0NTY3ODkwMTI=; hash of the body
b=dzdVyOfAKCdLXdJOc9G2q8LoXSlEniSbav+yuU4zGeeruD00lszZVoG4ZHRNiYzR

signature of the <headers in **h**> || previous fields of DKIM-Signature


Sending Mail Server
Private DKIM key
Send DKIM-signed Email
DKIM-SIGNATURE included in headers
Recieve Email
Receiving Mail Server
Extract dkim-signature from email headers
Fetch example.com DKIM record at 's._domainkey.example.com'
d-example.com s=s.domainkey
example.com DNS
Return example.com DKIM record
Validate message using public key from DKIM DNS entry
Was message unchanged?
Yes
No
DKIM Pass
DKIM Fail

[image credits]

DKIM: PROBLEMS

- Messages could be modified while in transit, potentially invalidating the signature. **MITIGATION**: DKIM defines **CANONICALIZATION** rules that allows to tolerate specific (small/cosmetic) changes to some header fields or the body content.

- **CANONICALIZATION rules may be not enough in some scenarios**, e.g., a mailing list is forwarding a message, modifying the subject and the content

- **DKIM does not provide confidentiality**

- **Domain listed in the DKIM Signature does need to be the same as the one in From: Why? Mailing list scenario: From: is user@gmail.com but the message is sent by another server (malinglist.com) which does not have the private key of gmail.com! It is up to the receiver to accept the message signed by third-party server.**

Domain-based Message Authentication, Reporting and Conformance (DMARC)

DMARC extends SPF and DKIM, allowing an organization to publish a policy that defines its email authentication practices and provides instructions to receiving mail servers for how to enforce them. In other words, DMARC allows a domain to say what to do with a message when DKIM/SPF fails and how to report abuses. The policy is published with a DNS TXT record.

Example:

_dmarc.example.com TXT v=DMARC1; p=reject; pct=100; rua=mailto:aggregate-reports@example.com;ruf=mailto:forensics-reports@example.com

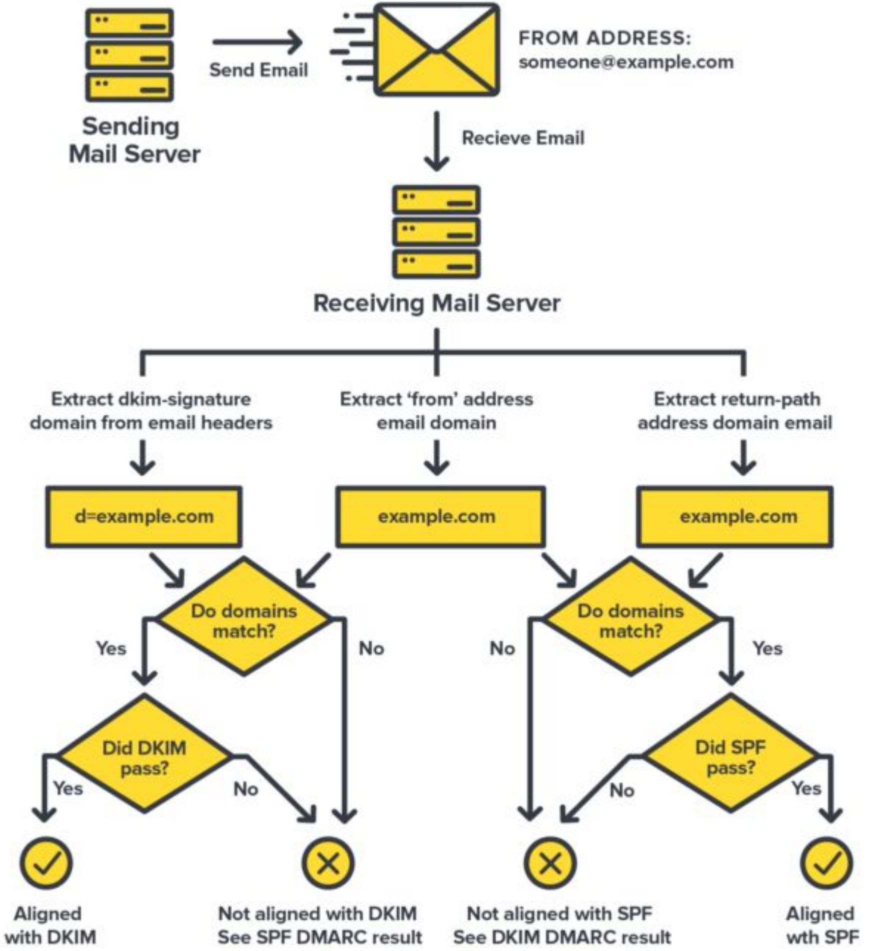
Check 100% of the messages, reject in case of failure, report failures to that address. Other policies may be: **none** (treat as without DMARC), **quarantine** (keep it but mark as spam)

Reference: Wikipedia DMARC

DMARC VALIDATION OUTCOME

Outcomes from the validation are reported in the mail headers:

```
Authentication-Results: mx.google.com;
    dkim=pass header.i=@enisa.europa.eu header.s=enisadkim header.b=B8Ea0tk+;
    spf=pass (google.com: domain of prokopios.drogkaris@enisa.europa.eu designates 139.91.222.30 as
permitted sender)
```



[image credits]

DMARC REPORTS

Reports are generated by inbound mail servers as part of the DMARC validation process.

There are two formats of DMARC reports:

- **Aggregate reports**: XML documents with statistical data about the messages form a domain. Data includes authentication results and message disposition. Aggregate reports are designed to be machine-readable. See here for an example.

- **Forensic reports**: individual copies of messages which failed authentication, each enclosed in a full email message using a special format called AFRF. Forensic report can be useful both for troubleshooting a domain's own authentication issues and for identifying malicious domains and web sites.

Authenticated Received Chain (ARC)

DKIM and SPF may break when a message is forwarded or altered by some SMTP servers. ARC provides a way to authenticate the entire chain traversed by a message, while SPF and DKIM authenticate only the original sender.

This is done by using additional headers:

- **ARC-Authentication-Results**: A combination of an instance number and the results of the SPF, DKIM, and DMARC validation
- **ARC-Seal**: A combination of an instance number, a DKIM-like signature of the previous ARC-Seal headers, and the validity of the prior ARC entries.
- **ARC-Message-Signature**: A combination of an instance number and a DKIM-like signature of the entire message except for the ARC-Seal headers

The basic idea is that each hop in the chain signs the message.

Reference: Wikipedia ARC

ARC: EXAMPLE (1)

```
Return-Path: <jqd@d1.example>
Received: from mail-ob0-f188.google.example
    (mail-ob0-f188.google.example [208.69.40.157]) by
    clochette.example.org with ESMTP id d200mr22663000ykb.93.1421363268
    for <fmartin@example.org>; Thu, 14 Jan 2015 15:03:15 -0800 (PST)
Received: from example.org (example.org [208.69.40.157])
    by gmail.example with ESMTP id d200mr22663000ykb.93.1421363207
    for <fmartin@example.com>; Thu, 14 Jan 2015 15:02:40 -0800 (PST)
Received: from segv.d1.example (segv.d1.example [72.52.75.15])
    by lists.example.org (8.14.5/8.14.5) with ESMTP id t0EKaNU9010123
    for <arc@example.org>; Thu, 14 Jan 2015 15:01:30 -0800 (PST)
    (envelope-from jqd@d1.example)
Received: from [2001:DB8::1A] (w-x-y-z.dsl.static.isp.example [w.x.y.z])
    (authenticated bits=0)
    by segv.d1.example with ESMTP id t0FN4a80084569;
    Thu, 14 Jan 2015 15:00:01 -0800 (PST)
    (envelope-from jqd@d1.example)
```

The message has traversed 4 hops

Example taken from here

ARC: EXAMPLE (2)

```
ARC-Seal: i=3; a=rsa-sha256; cv=pass; d=clochette.example.org; s=
        clochette; t=12345; b=CU87XzXlNlk5X/yW4l73UvPUcP9ivwYWxyBWcVrRs7
        +HPx3K05nJhny2fvymbReAmOA9GTH/y+k9kEc59hAKVg==
ARC-Message-Signature: i=3; a=rsa-sha256; c=relaxed/relaxed; d=
        clochette.example.org; h=message-id:date:from:to:subject; s=
        clochette; t=12345; bh=KWSe46TZKCcDbH4klJPo+tjk5LWJnVRlP5pvjXFZY
        LQ=; b=o71vwyLsK+Wm4cOSlirXoRwzEvi0vqIjd/2/GkYFYlSd/GGfKzkAgPqxf
        K7ccBMP7Zjb/mpeggswHjEMS8x5NQ==
ARC-Authentication-Results: i=3; clochette.example.org; spf=fail
    smtp.from=jqd@d1.example; dkim=fail (512-bit key)
    header.i=@d1.example; dmarc=fail; arc=pass (as.2.gmail.example=pass,
    ams.2.gmail.example=pass, as.1.lists.example.org=pass,
    ams.1.lists.example.org=fail (message has been altered))
Authentication-Results: clochette.example.org; spf=fail
    smtp.from=jqd@d1.example; dkim=fail (512-bit key)
    header.i=@d1.example; dmarc=fail; arc=pass (as.2.gmail.example=pass,
    ams.2.gmail.example=pass, as.1.lists.example.org=pass,
    ams.1.lists.example.org=fail (message has been altered))
```

i=3:

- **ARC-Seal**
- **ARC-Message-Signature**
- **ARC-Authentication-Results**

Since this is the last, then we have also the final Authentication-Results

ARC: EXAMPLE (3)

```
ARC-Seal: i=2; a=rsa-sha256; cv=pass; d=gmail.example; s=20120806; t=
      12345; b=Zpukh/kJL4Q7Kv391FKwTepgS56dgHIcdhhJZjsalhqkFIQQAJ4T9BE
      8jjLXWpRNuh81yqnT1/jHn086RwezGw==
ARC-Message-Signature: i=2; a=rsa-sha256; c=relaxed/relaxed; d=
      gmail.example; h=message-id:date:from:to:subject; s=20120806; t=
      12345; bh=KWSe46TZKCcDbH4klJPo+tjk5LWJnVRlP5pvjXFZYLQ=; b=CVoG44
cVZvoSs2mMig2wwqPaJ4OZS5XGMCegWqQs1wvRZJS894tJM0xO1RJLgCPsBOxdA59WSqI9s9DfyKDfWg==
ARC-Authentication-Results: i=2; gmail.example; spf=fail
   smtp.from=jqd@d1.example; dkim=fail (512-bit key)
   header.i=@example.org; dmarc=fail; arc=pass
   (as.1.lists.example.org=pass, ams.1.lists.example.org=pass)
```

i=2:

- **ARC-Seal**
- **ARC-Message-Signature**
- **ARC-Authentication-Results**

ARC: EXAMPLE (4)

ARC-Seal: i=1; a=rsa-sha256; **cv=none**; d=lists.example.org; s=dk-lists;
t=12345; b=TlCCKzgk3TrAa+G77gYYO8Fxk4q/Ml0biqduZJeOYh6+0zhwQ8u/
lHxLi21pxu347isLSuNtvIagIvAQna9a5A==
ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed; d=
lists.example.org; h=message-id:date:from:to:subject; s=
dk-lists; t=12345; bh=KWSe46TZKCcDbH4klJPo+tjk5LWJnVRlP5pvjXFZYL
Q=; b=DsoD3n3hiwlrN1ma8IZQFgZx8EDO7Wah3hUjIEsYKuShRKYB4LwGUiKD5Y
yHgcIwGHhSc/4+ewYqHMWDnuFxiQ==
ARC-Authentication-Results: i=1; **lists.example.org**; spf=pass
smtp.from=jqd@d1.example; dkim=pass (512-bit key)
header.i=@d1.example; dmarc=pass

i=1:

- **ARC-Seal**
- **ARC-Message-Signature**
- **ARC-Authentication-Results**