Proton Mail Account Age Lookup: What a Public PGP Endpoint Can Reveal

Maria Cattini | 27/08/2026 | CYBERSECURITY

An anonymous email address rarely tells an investigator much by itself. A Proton Mail address can be created without exposing the identity behind it, and knowing the address does not provide access to correspondence, credentials or private account data.

There is, however, a narrower piece of information that can sometimes be recovered from public infrastructure: a timestamp associated with the account's public PGP key.

The method documented in the supplied material uses a public Proton Mail key lookup endpoint. It requires no authentication, password guessing or access to private communications. Its investigative value is temporal: it can help establish whether an account appears to have existed long before an event under investigation or was created close to it.

That is a modest result. In an OSINT investigation, modest results can still change the chronology of a case.

What the technique actually retrieves

The workflow starts from an email address and queries Proton Mail's public PGP infrastructure. According to the supplied material, the technique works for most active Proton Mail accounts unless the user has disabled the public PGP key, which is described as enabled by default.

The example provided is:

gabaigaduka@protonmail.com

The public endpoint shown in the source is:

https://api.protonmail.ch/pks/lookup?op=index&search=gabaigaduka@protonmail.com

The response contains several fields. The relevant entry begins with pub::

pub:cc0322777176da3a4a4d9916e4f32c96c55663a4:1:2048:1527584191::

The value 1527584191 is a Unix timestamp: a time represented as the number of seconds elapsed since 1 January 1970. The same timestamp also appears in the uid: line associated with the email address.

At this point, the useful output is not an identity. It is a temporal marker.

Testing the workflow

The lookup itself is simple enough to reproduce from a browser or through curl, as described in the source.

The investigative sequence matters more than the technical complexity.

First, start with the Proton Mail address being examined and submit it to the public PGP lookup endpoint using the op=index operation and the address as the search parameter.

Second, inspect the returned data rather than treating the entire response as meaningful. The source specifically identifies the pub: record and its Unix timestamp as the field of interest.

Third, convert that numeric value into a human-readable date. The supplied example uses EpochConverter, although the source notes that another Unix timestamp converter can perform the same task.

For 1527584191, the conversion produces 29 May 2018 at approximately 15:36 UTC. The source therefore concludes that the example account dates from late spring 2018.

The operational result is a date associated with the public-key metadata exposed for that account.

What happens next is where OSINT methodology becomes more important than the lookup.

Where the result becomes useful

An account date has little meaning in isolation. Its value appears when it is placed on a timeline containing independently established events.

Consider the investigative distinction described in the source. If an account associated with an incident appears to have been created one hour after that incident, the timing becomes an observation worth documenting. If the same address had already existed for several years, the chronology points in a different direction.

Neither result identifies the account owner.

Nor does temporal proximity prove that an account was created for a specific operation. A timestamp can establish chronology; it cannot establish motive.

This is precisely where a small technical lookup can be mishandled. The temptation is to turn correlation into attribution: account created after event, therefore account created because of event. The source does not support that inference.

A stronger workflow records the timestamp as one piece of evidence and compares it with other independently verified observations.

What the tool does not reveal

The technique has a deliberately narrow scope.

It does not recover the account password. It does not expose correspondence. It does not provide access to the mailbox. The source describes the information obtained as technical metadata available through a lawful request to Proton Mail’s public API infrastructure.

It also has an availability constraint. According to the supplied material, the method depends on the account having a public PGP key available. If that public key has been disabled, the technique may not produce the expected result.

That limitation changes how a failed query should be interpreted.

Failure to retrieve the metadata is not evidence that an account is fake, recently created or inactive. It means only that the expected information was not obtained through this particular lookup. Treating absence of a result as evidence about the person behind the address would go beyond what the method can establish.

The OSINT value is chronology, not attribution

This technique illustrates a recurring pattern in open-source investigations: publicly exposed infrastructure can contain metadata that was not designed primarily for investigative use but becomes informative when placed in context.

The endpoint does not answer the question investigators are usually most interested in: who controls this address?

It answers a smaller question: what temporal information can be recovered from the public PGP metadata associated with it?

That distinction prevents the tool from being oversold.

Used correctly, the lookup can contribute to timeline reconstruction. An analyst can record the email address examined, preserve the relevant response, extract the timestamp, convert it to UTC and compare that time with other documented events.

The evidentiary chain remains visible: public endpoint, returned metadata, timestamp conversion, chronological comparison.

What comes after that is interpretation.

A close temporal relationship can justify further investigation. It cannot, by itself, identify an operator, establish coordination or prove why the account was created.

That boundary is what makes the method useful rather than misleading.

ProjectOSINT assessment

Function: Retrieve publicly exposed PGP metadata associated with a Proton Mail address and extract the timestamp described in the supplied material.

Best use: Timeline analysis during an investigation involving a known Proton Mail address.

Input: Proton Mail email address.

Output: Public PGP lookup information, including the Unix timestamp identified in the source.

Main limitation: The workflow depends on the relevant public PGP key being available.

Verification requirement: Preserve the original lookup result and distinguish the retrieved timestamp from any later interpretation based on its relationship to other events.

Analytical risk: Treating chronological proximity as evidence of identity, intent or causation.

Verdict: A narrow but potentially useful OSINT technique. Its value comes from adding a temporal reference to an investigation, not from deanonymising the person behind the account.
An anonymous email address rarely tells an investigator much by itself. A Proton Mail address can be created without exposing the identity behind it, and knowing the address does not provide access to correspondence, credentials or private account data.

There is, however, a narrower piece of information that can sometimes be recovered from public infrastructure: a timestamp associated with the account’s public PGP key.

The method documented in the supplied material uses a public Proton Mail key lookup endpoint. It requires no authentication, password guessing or access to private communications. Its investigative value is temporal: it can help establish whether an account appears to have existed long before an event under investigation or was created close to it.

That is a modest result. In an OSINT investigation, modest results can still change the chronology of a case.

What the technique actually retrieves

The workflow starts from an email address and queries Proton Mail's public PGP infrastructure. According to the supplied material, the technique works for most active Proton Mail accounts unless the user has disabled the public PGP key, which is described as enabled by default.

The example provided is:

gabaigaduka@protonmail.com

The public endpoint shown in the source is:

https://api.protonmail.ch/pks/lookup?op=index&search=gabaigaduka@protonmail.com

The response contains several fields. The relevant entry begins with pub::

pub:cc0322777176da3a4a4d9916e4f32c96c55663a4:1:2048:1527584191::

The value 1527584191 is a Unix timestamp: a time represented as the number of seconds elapsed since 1 January 1970. The same timestamp also appears in the uid: line associated with the email address.

At this point, the useful output is not an identity. It is a temporal marker.

Testing the workflow

The lookup itself is simple enough to reproduce from a browser or through curl, as described in the source.

The investigative sequence matters more than the technical complexity.

First, start with the Proton Mail address being examined and submit it to the public PGP lookup endpoint using the op=index operation and the address as the search parameter.

Second, inspect the returned data rather than treating the entire response as meaningful. The source specifically identifies the pub: record and its Unix timestamp as the field of interest.

Third, convert that numeric value into a human-readable date. The supplied example uses EpochConverter, although the source notes that another Unix timestamp converter can perform the same task.

For 1527584191, the conversion produces 29 May 2018 at approximately 15:36 UTC. The source therefore concludes that the example account dates from late spring 2018.

The operational result is a date associated with the public-key metadata exposed for that account.

What happens next is where OSINT methodology becomes more important than the lookup.

Where the result becomes useful

An account date has little meaning in isolation. Its value appears when it is placed on a timeline containing independently established events.

Consider the investigative distinction described in the source. If an account associated with an incident appears to have been created one hour after that incident, the timing becomes an

observation worth documenting. If the same address had already existed for several years, the chronology points in a different direction.

Neither result identifies the account owner.

Nor does temporal proximity prove that an account was created for a specific operation. A timestamp can establish chronology; it cannot establish motive.

This is precisely where a small technical lookup can be mishandled. The temptation is to turn correlation into attribution: account created after event, therefore account created because of event. The source does not support that inference.

A stronger workflow records the timestamp as one piece of evidence and compares it with other independently verified observations.

What the tool does not reveal

The technique has a deliberately narrow scope.

It does not recover the account password. It does not expose correspondence. It does not provide access to the mailbox. The source describes the information obtained as technical metadata available through a lawful request to Proton Mail’s public API infrastructure.

It also has an availability constraint. According to the supplied material, the method depends on the account having a public PGP key available. If that public key has been disabled, the technique may not produce the expected result.

That limitation changes how a failed query should be interpreted.

Failure to retrieve the metadata is not evidence that an account is fake, recently created or inactive. It means only that the expected information was not obtained through this particular lookup. Treating absence of a result as evidence about the person behind the address would go beyond what the method can establish.

The OSINT value is chronology, not attribution

This technique illustrates a recurring pattern in open-source investigations: publicly exposed infrastructure can contain metadata that was not designed primarily for investigative use but becomes informative when placed in context.

The endpoint does not answer the question investigators are usually most interested in: who controls this address?

It answers a smaller question: what temporal information can be recovered from the public PGP metadata associated with it?

That distinction prevents the tool from being oversold.

Used correctly, the lookup can contribute to timeline reconstruction. An analyst can record the email address examined, preserve the relevant response, extract the timestamp, convert it to UTC and compare that time with other documented events.

The evidentiary chain remains visible: public endpoint, returned metadata, timestamp conversion, chronological comparison.

What comes after that is interpretation.

A close temporal relationship can justify further investigation. It cannot, by itself, identify an operator, establish coordination or prove why the account was created.

That boundary is what makes the method useful rather than misleading.

ProjectOSINT assessment

Function: Retrieve publicly exposed PGP metadata associated with a Proton Mail address and extract the timestamp described in the supplied material.

Best use: Timeline analysis during an investigation involving a known Proton Mail address.

Input: Proton Mail email address.

Output: Public PGP lookup information, including the Unix timestamp identified in the source.

Main limitation: The workflow depends on the relevant public PGP key being available.

Verification requirement: Preserve the original lookup result and distinguish the retrieved timestamp from any later interpretation based on its relationship to other events.

Analytical risk: Treating chronological proximity as evidence of identity, intent or causation.

Verdict: A narrow but potentially useful OSINT technique. Its value comes from adding a temporal reference to an investigation, not from deanonymising the person behind the account.