Punycode Attack

`Punycode is a way of converting words that cannot be written in ASCII, into a Unicode ASCII encoding.`

- attackers redirect users to a malicious domain that seems legitimate at first glance.
- `adıdas.de` which has the Punycode of `http://xn--addas-o4a.de/`
- Attackers usually hide the malicious domains under **URL Shorteners**
  A URL Shortener is a tool that creates a short and unique URL that will redirect to the specific website specified during the initial step of setting up the URL Shortener link

Why is it so hard to detect this kind of fraud?

What does a given URL look like and how come we can so easily confuse it? It can be written, for example, in Cyrillic, i.e., with different characters, which at first glance look very similar to the symbols we use.

Look at these simple examples:

(service → legit domain → fake domain)

- Microsoft → microsoft.com → mìcrosoft[.]com
- Paypal → paypal.com → paypaḷ[.]com
- Coinbase → coinbase.com → ćoinbase[.]name
- Blockchain → blockchain.com → blóckchäin[.]com
- Twitter → twitter.com → twittër[.]com
- Rolex ➡️ rołex.com.

How can you defend yourself from homograph attacks?

Protection depends primarily on the caution of individual users. Always watch out for e-mails that want you to do something fast. Always check the links carefully to see if there is a slightly different letter in it (see the examples above, i.e., *paypaḷ*). If you have received an e-mail with a suspicious offer from a company, you should first check their official website to see if there is any information there about a similar offer. The easiest protection is not to click on the link at all, because it is usually fake.

Punycode can also be used to create a fake corporate email. If your management writes to you that he or she wants to send a certain amount of money to some account, carefully look at his email address if there are any flaws to be found. Or just copy it to the address book if it has a match in it. If there is a different but similar letter, the address book will see the difference.

We have an extensive database of malicious domains at Whalebone. That database reveals with precision if that is a secure link or not. And if it's the latter, then it does not allow the end user to enter the potentially risky page, thus preventing the associated risks.