How many Mastodon instances are out there?

Ever since I spun up my own Mastodon server and I've been watching my web logs just to see all the other unique instances that connect to mine. My personal server with only one user currently knows about ~1800 other servers. But there's got to be more than that out there, right? I thought so too. So I wrote a fairly simple bit of PowerShell that lets me crawl the reference tree from one server to the next and try and count all the instances on the internet.

Get-MastodonInstances.ps1

```
$instances = New-Object System.Collections.Generic.HashSet[string] -ArgumentList
([StringComparer]::OrdinalIgnoreCase)
$erroredInstances = New-Object System.Collections.Generic.HashSet[string] -ArgumentList
([StringComparer]::OrdinalIgnoreCase)
$newInstances = New-Object System.Collections.Generic.List[string]

$instances.Add("mastodon.social")
$newInstances.Add("mastodon.social")

for ($i = 0; $i -lt $newInstances.Count; $i++) {
    $instance = $newInstances[$i]
    Write-Host "$($i + 1) / $($newInstances.Count) ($instance)"

    try {
        $response = Invoke-WebRequest `
            -Uri "https://$instance/api/v1/instance/peers" `
            -Method GET `
            -ErrorAction SilentlyContinue
    }
    catch {
        $erroredInstances.add($instance) | Out-Null
        continue
```

```
    }

    if ([string]::IsNullOrWhiteSpace($response.Content)) {
        continue
    }

    $response.Content |
    ConvertFrom-Json |
    Where-Object { ($_.IndexOf("*") -lt 0) -and $instances.Add($_) } |
    ForEach-Object { $newInstances.Add($_.ToString()) }
}

$instances | Out-File -FilePath "MastodonInstances.txt"
$erroredInstances | Out-File -FilePath "MastodonErroredInstances.txt"

$instances.Count
```

There are some caveats to this approach, of course. Some of the instances the script encountered were offline, some were not actually Mastodon, but other implementors of ActivityPub (Friendica, Pleroma, etc.), and some just didn't like me querying them anonymously. There is also the case where certain "social circles" have completely cut themselves off from the rest of the Fediverse. I'll never know about those servers.

My script has been running for about an hour now, and it's only now querying the ~2700th host, and has added **~44,000** to the backlog. It'll be a while before it finishes. Offline domains slow it down a lot while waiting for the timeout. I suppose I could have worked in multi-threading, but this was just a quick-and-dirty experiment. I'll update this post if and when it ever comes up with a final number.

Update

I had to restart the script because when I came back in the morning, it had maxed out my laptop's memory. I've slightly optimized the script by adding to a .NET `System.Collections.Generic.List<string>` rather than re-creating a PowerShell array each iteration through the loop. Also added a try/catch block to filter out the errors I was seeing in the console and just to keep track of them.

Luckily, it's not very taxing to run this script because most of the run time is waiting for the other servers to respond.

Revision #4
Created 2022-11-28 16:44:15 AST by Steve Dinn
Updated 2023-05-01 19:33:08 ADT by Steve Dinn