Go Domain Email Security Checker - Project Overview

Project Summary

This Go project is a command-line tool designed to check whether a given domain has configured email security records: MX, SPF, and DMARC.

The user types domain names into standard input (keyboard or piped text), and for each domain, the program checks DNS records and prints results in CSV format.

Code Flow Explanation

1. The program starts with `main()` and uses `bufio.Scanner` to read domain names from standard input.
2. Each input line is passed to the `checkDomain()` function.
3. Inside `checkDomain()`:
   - It checks for MX records using `net.LookupMX`.
   - It checks for SPF by looking for TXT records starting with 'v=spf1'.
   - It checks for DMARC by looking for TXT records starting with 'v=DMARC1' under the '_dmarc.' subdomain.
4. Results are printed in a comma-separated format with six columns:
   domain, hasMX, hasSPF, spfRecord, hasDMARC, dmarcRecord

How to Use

Usage Instructions:

1. Run the program from the terminal.
2. Type or pipe domain names into it (e.g., 'google.com').
3. It will output results line by line in a CSV-friendly format.

Example:

$ go run main.go

Go Domain Email Security Checker - Project Overview

google.com

apple.com

Output:

domain, hasMX, hasSPF, spfRecord, hasDMARC, dmarcRecord

google.com, true, true, v=spf1 ..., true, v=DMARC1 ...

apple.com, true, true, v=spf1 ..., true, v=DMARC1 ...

Source Code

```
package main

import (
 "fmt"
 "bufio"
 "log"
 "net"
 "os"
 "strings"
)

func main() {
 scanner := bufio.NewScanner(os.Stdin)
 fmt.Println("domain, hasMX, hasSPF, spfRecord, hasDMARC, dmarcRecord")

 for scanner.Scan() {
  checkDomain(scanner.Text())
  //whatever you type goes into checkDomain function

 }

 // Check for errors after scanning
 // This is important to ensure we handle any issues with reading input.
 if err:= scanner.Err(); err!=nil {
  log.Fatal("Error:could not read from input:", err) //for non-recoverable errors
  // This will log the error and exit the program.
 }
}
```

Go Domain Email Security Checker - Project Overview

```
func checkDomain(domain string) {
 var hasMX, hasSPF, hasDMARC bool //declaration of variables
 var spfRecord, dmarcRecord string

 mxRecords, err := net.LookupMX(domain) // Check for MX records

 if err != nil {
  log.Printf("Error: %v\n", err) // for recoverable errors
 }

 if len(mxRecords) > 0 {
  hasMX = true
 }

 txtRecords, err := net.LookupTXT(domain) // Check for TXT records

 if err != nil {
  log.Printf("Error : %v\n", err)
 }

 for _, record := range txtRecords {
  if strings.HasPrefix(record, "v=spf1") {
   hasSPF = true
   spfRecord = record // Store the SPF record
   break
  }
 }

 dmarcRecords , err := net.LookupTXT("_dmarc." + domain) // Check for DMARC records

 if err != nil {
  log.Printf("Error: %v\n", err)
 }

 for _, record := range dmarcRecords {
  if strings.HasPrefix(record, "v=DMARC1"){
   hasDMARC = true
   dmarcRecord = record // Store the DMARC record
   break
  }
 }
```

Go Domain Email Security Checker - Project Overview

```
 fmt.Printf("%v, %v, %v, %v, %v, %v", domain, hasMX, hasSPF, spfRecord, hasDMARC, dmarcRecord)
}
```