



# Documentação das Funções

## log\_message(message)

Registra mensagens com timestamps no arquivo de log.

### Parâmetros:

- `message` (str): Mensagem a ser registrada no log.

### Comportamento:

- Abre o arquivo de log (LOG\_FILE) em modo de anexação.
- Escreve a mensagem fornecida com um timestamp.

### Exemplo de Uso:

```
log_message("Início da atualização do DuckDNS.")
```

## mostrar\_loading(message, duration=6)

Mostra uma animação de loading no console.

### Parâmetros:

- `message` (str): Mensagem a ser exibida antes da animação de loading.
- `duration` (int, opcional): Duração da animação em segundos (padrão é 6 segundos).

### Comportamento:

- Exibe uma animação de loading com caracteres em movimento.
- Atualiza o console a cada intervalo de tempo até que o tempo especificado expire.

**Exemplo de Uso:**

```
mostrar_loading("Atualizando DuckDNS", duration=2)
```

## mostrar\_banner()

Exibe o banner do script.

**Comportamento:**

- Imprime um banner colorido no console usando caracteres ASCII.

**Exemplo de Uso:**

```
mostrar_banner()
```

## get\_vpn\_ip()

Obtém o IP do adaptador VPN.

**Retorna:**

- `str` ou `None`: Endereço IP do adaptador VPN, ou None se não for possível obter o IP.

**Comportamento:**

- Itera sobre as interfaces de rede disponíveis e verifica se alguma delas é uma interface VPN (identificada por 'tun' ou 'vpn' no nome).
- Retorna o IP associado à interface VPN.

**Exemplo de Uso:**

```
ip_vpn = get_vpn_ip()
```

## atualiza\_duckdns(domain, ip\_interno, token)

Atualiza o DuckDNS com o IP atual.

### **Parâmetros:**

- `domain` (str): Nome do domínio DuckDNS a ser atualizado.
- `ip_interno` (str): IP interno que será configurado para o domínio.
- `token` (str): Token de autenticação do DuckDNS.

### **Comportamento:**

- Faz uma solicitação HTTP GET para o serviço DuckDNS para atualizar o IP.
- Exibe uma mensagem de sucesso ou erro com base na resposta.

### **Exemplo de Uso:**

```
atualiza_duckdns("teste-123", "192.168.1.1", "124232")
```

## **teste\_icmp(domain)**

Testa a conexão ICMP (ping) com o domínio DuckDNS.

### **Parâmetros:**

- `domain` (str): Nome do domínio DuckDNS a ser testado.

### **Comportamento:**

- Realiza um ping no domínio DuckDNS.
- Exibe uma mensagem indicando se o ping foi bem-sucedido ou não.

### **Exemplo de Uso:**

```
teste_icmp("teste-123")
```

## **config\_json(token, domain, ip\_interno)**

Salva a configuração no arquivo JSON.

### **Parâmetros:**

- `token` (str): Token de autenticação do DuckDNS.
- `domain` (str): Nome do domínio DuckDNS.
- `ip_interno` (str): IP interno configurado.

### **Comportamento:**

- Cria ou atualiza o arquivo JSON de configuração (CONFIG\_FILE) com o token, domínio e IP interno fornecidos.

**Exemplo de Uso:**

```
config_json("124232", "teste-123", "192.168.1.1")
```

## carregar\_config()

Carrega a configuração do arquivo JSON.

**Retorna:**

- `dict` ou `None` : Configuração carregada a partir do arquivo JSON, ou None se o arquivo não existir ou não puder ser lido.

**Comportamento:**

- Lê o arquivo JSON de configuração e retorna o conteúdo como um dicionário.

**Exemplo de Uso:**

```
config = carregar_config()
```

## verificar\_e\_atualizar()

Verifica o IP atual e atualiza o DuckDNS se necessário.

**Comportamento:**

- Carrega a configuração existente.
- Obtém o IP atual da VPN.
- Compara o IP atual com o IP configurado.
- Atualiza o DuckDNS se o IP tiver mudado e salva a nova configuração.

**Exemplo de Uso:**

```
verificar_e_atualizar()
```

## verificar\_dns(domain, ip\_atual)

Verifica se o DNS está apontando para o IP correto.

**Parâmetros:**

- `domain` (str): Nome do domínio DuckDNS.
- `ip_atual` (str): IP atual para o qual o domínio deve apontar.

**Retorna:**

- `bool`: True se o DNS estiver configurado corretamente, False caso contrário.

**Comportamento:**

- Realiza uma consulta DNS para verificar se o domínio DuckDNS está apontando para o IP correto.
- Retorna True se o IP retornado pela consulta DNS corresponder ao IP atual, False caso contrário.

**Exemplo de Uso:**

```
dns_correto = verificar_dns("teste-123", "192.168.1.1")
```