


```
workingPath, _ := os.Getwd()&#xA;    realPath := filepath.Join(workingPath, path)&#xA;    stat, err
:= os.Stat(realPath)&#xA;    if err != nil {&#xA;        if os.IsNotExist(err) {&#xA;
log.Println(realPath + &#34; does not exist&#34;)&#xA;        return&#xA;    }&#xA;    }&#xA;
if stat.IsDir() {&#xA;        realPath = filepath.Join(realPath, &#34;index.html&#34;)&#xA;        if stat,
err = os.Stat(realPath); err != nil {&#xA;            if os.IsNotExist(err) {&#xA;
log.Println(realPath + &#34; not found&#34;)&#xA;            }&#xA;        }&#xA;    }&#xA;
log.Println(&#34;serve file &#34; + realPath)&#xA;    http.ServeFile(w, r, realPath)&#xA;
}&#xA;}Der Aufbau entspricht einem normalen HTTP Server über die http.Server Instanz. Dieser
wird mit dem folgenden Befehl zu einem HTTP/2 Server:http2.ConfigureServer(srv,
&amp;http2.Server{})Die HTTP/2 Push Funktion erfolgt über die http.Pusher Abfrage. Solche ist ok
true wenn der Server dies unterstützt:pusher, ok := w.(http.Pusher)&#xA;if ok {&#xA;    options :=
&amp;http.PushOptions{&#xA;        Header: http.Header {&#xA;            &#34;Accept-Encoding&#34;:
r.Header[&#34;Accept-Encoding&#34;],&#xA;        },&#xA;    }&#xA;
pusher.Push(&#34;/css/main.css&#34;,, options)&#xA;}Das Accept-Encoding ist wichtig, wenn z.B.
die GZip Unterstützung benötigt wird.
```

Feedback

War dieser Blog für Sie wertvoll. Wir danken für jede Anregung und Feedback

Kontakt

Simtech AG
Finkenweg 23
3110 Münsingen
Schweiz

Impressum

Das Copyright für sämtliche Inhalte dieser Website liegt bei Simtech AG, Schweiz.
Beachten Sie auch unsere Hinweise zum Urheberrecht, Datenschutz und Haftungsausschluss.
Jeder Hinweis auf Fehler nehmen wir gerne entgegen.

Copyright

2024 Simtech AG, All rights reserved, Powered by stack.ch written in Golang by Daniel Schmutz

<https://www.simtech-ag.ch/Golang Blogs 2 Push Go HTTP>