Code As Infrastructure

559 rules found
311 error prone
248 security issues
code health checked = A+
Snyx
Codacy
Code Factor
GitHub
Shopify - My Businesses, Networks, devices, accounts, emails

Synk - GitHub
https://app.snyk.io/org/pharaoh911/manage/cloud-config
 Code Factor
https://www.codefactor.io/repository/github/pharaoh911/lines-of-code
Cody
https://app.codacy.com/gh/Pharaoh911/lines-of-code/patterns/list
GitHub

TYPE
- CloudFormation

Terraform

API Gateway (REST APIs)
SNYK-CC-AWS-419
Open this link in a new tab
API gateway cached responses are not encrypted
HighMediumLowNone

SNYK-CC-AWS-452

Open this link in a new tab

Wildcard action in api gateway access policy

HighMediumLowNone

SNYK-CC-AWS-453

Open this link in a new tab

Wildcard principal in rest api access policy

HighMediumLowNone

SNYK-CC-TF-63

Open this link in a new tab

API Gateway does not enforce latest TLS/SSL policy

HighMediumLowNone

SNYK-CC-TF-99

Open this link in a new tab

API Gateway allows anonymous access

HighMediumLowNone

SNYK-CC-TF-129

Open this link in a new tab

API Gateway X-Ray tracing disabled

HighMediumLowNone

SNYK-CC-TF-138

Open this link in a new tab

API Gateway access logging disabled

HighMediumLowNone

Athena

SNYK-CC-TF-110

Open this link in a new tab

Non-encrypted Athena DB at rest

HighMediumLowNone

SNYK-CC-TF-113

Open this link in a new tab

Athena workgroup does not enforce service settings

HighMediumLowNone

Batch

SNYK-CC-AWS-430

Open this link in a new tab

Batch job runs in privileged mode

HighMediumLowNone

CloudFront

SNYK-CC-TF-57

Open this link in a new tab

Cloudfront distribution does not enforce HTTPS

HighMediumLowNone

SNYK-CC-TF-58

Open this link in a new tab

Distribution does not enforce latest TLS/SSL policy

HighMediumLowNone

SNYK-CC-TF-75

Open this link in a new tab

Cloudfront distribution without WAF

HighMediumLowNone

SNYK-CC-TF-142

Open this link in a new tab

CloudFront access logging disabled

HighMediumLowNone

CloudTrail

SNYK-CC-TF-15

Open this link in a new tab

CloudTrail has logging disabled

HighMediumLowNone

SNYK-CC-TF-16

CloudTrail does not have log file validation enabled

HighMediumLowNone

SNYK-CC-TF-17

CloudTrail logs are not encrypted with managed key

HighMediumLowNone

SNYK-CC-TF-135

CloudTrail does not include all regions

HighMediumLowNone

SNYK-CC-TF-256

CloudTrail not integrated with CloudWatch

HighMediumLowNone

CloudWatch

SNYK-CC-AWS-415

CloudWatch log group not encrypted with managed key

HighMediumLowNone

SNYK-CC-TF-134

CloudWatch Log group retention period not set

HighMediumLowNone

CodeBuild

SNYK-CC-TF-111

Non-Encrypted CodeBuild artifacts

HighMediumLowNone

Cognito

SNYK-CC-AWS-451

Open this link in a new tab

Cognito user pool without MFA

HighMediumLowNone

Config

SNYK-CC-AWS-406

Open this link in a new tab

Configuration aggregator does not contain all regions

HighMediumLowNone

DocumentDB

SNYK-CC-AWS-416

Open this link in a new tab

Docdb cluster not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-AWS-435

Open this link in a new tab

Audit logging is not enabled in DocDB parameter group

HighMediumLowNone

SNYK-CC-TF-109

Open this link in a new tab

Non-encrypted DocDB at rest

HighMediumLowNone

SNYK-CC-TF-141

Open this link in a new tab

DocDB logging disabled

HighMediumLowNone

DynamoDB

SNYK-CC-AWS-417

Open this link in a new tab

DynamoDB not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-TF-106

Open this link in a new tab

Non-encrypted DynamoDB DAX

HighMediumLowNone

SNYK-CC-TF-125

Open this link in a new tab

DynamoDB point-in-time recovery disabled

HighMediumLowNone

EBS

SNYK-CC-AWS-403

Open this link in a new tab

EBS encryption by default is disabled

HighMediumLowNone

EC2

SNYK-CC-AWS-402

Open this link in a new tab

AMI snapshot is not encrypted

HighMediumLowNone

SNYK-CC-AWS-404

Open this link in a new tab

AMI snapshot copy is not encrypted

HighMediumLowNone

SNYK-CC-AWS-424

Open this link in a new tab

Address source/destination check disabled on the instance

HighMediumLowNone

SNYK-CC-AWS-425

Open this link in a new tab

EC2 is unable to replace unhealthy instances

HighMediumLowNone

SNYK-CC-AWS-426

Open this link in a new tab

EC2 API termination protection is not enabled

HighMediumLowNone

SNYK-CC-AWS-429

Open this link in a new tab

Traffic mirroring session enabled

HighMediumLowNone

SNYK-CC-TF-2

Open this link in a new tab

Non-encrypted EBS snapshot

HighMediumLowNone

SNYK-CC-TF-3

Open this link in a new tab

Non-encrypted EBS volume

HighMediumLowNone

SNYK-CC-TF-46

Open this link in a new tab

AWS EC2-Classic resource detected

HighMediumLowNone

SNYK-CC-TF-51

Open this link in a new tab

Resource has public IP assigned

HighMediumLowNone

SNYK-CC-TF-53

Open this link in a new tab

Non-Encrypted root block device

HighMediumLowNone

SNYK-CC-TF-123

Open this link in a new tab

Hard coded secrets in EC2 metadata

HighMediumLowNone

SNYK-CC-TF-130

Open this link in a new tab

EC2 instance accepts IMDSv1

HighMediumLowNone

ECR

SNYK-CC-AWS-418

Open this link in a new tab

ECR repository is not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-TF-61

Open this link in a new tab

ECR image scanning is disabled

HighMediumLowNone

SNYK-CC-TF-93

Open this link in a new tab

ECR policy allows public access

HighMediumLowNone

SNYK-CC-TF-126

Open this link in a new tab

ECR Registry allows mutable tags

HighMediumLowNone

ECS

SNYK-CC-AWS-413

Open this link in a new tab

EFS in task definition does not encrypt data in transit

HighMediumLowNone

SNYK-CC-TF-52

Open this link in a new tab

Potentially sensitive variable in task definition

HighMediumLowNone

SNYK-CC-TF-128

ECS ContainerInsights disabled

HighMediumLowNone

EFS

SNYK-CC-TF-204

Non-encrypted EFS at rest

HighMediumLowNone

EKS

SNYK-CC-TF-94

EKS cluster allows public access

HighMediumLowNone

SNYK-CC-TF-107

EKS cluster has non-encrypted secrets at rest

HighMediumLowNone

SNYK-CC-TF-131

EKS control plane logging insufficient

HighMediumLowNone

Elastic Load Balancing

SNYK-CC-AWS-405

ALB does not drop invalid headers

HighMediumLowNone

SNYK-CC-TF-47

Load balancer endpoint does not enforce HTTPS

HighMediumLowNone

SNYK-CC-TF-48

Load balancer is internet facing

HighMediumLowNone

SNYK-CC-TF-49

ELB does not enforce latest TLS/SSL policy

HighMediumLowNone

ElastiCache

SNYK-CC-AWS-407

ElastiCache automatic backup is disabled

HighMediumLowNone

SNYK-CC-TF-71

Non-Encrypted ElastiCache Replication Group

HighMediumLowNone

SNYK-CC-TF-214

Non-Encrypted ElastiCache data in transit

HighMediumLowNone

SNYK-CC-TF-215

ElastiCache cluster does not require authentication

HighMediumLowNone

ElasticSearch

SNYK-CC-AWS-431

Wildcard action in ElasticSearch access policy

HighMediumLowNone

SNYK-CC-AWS-432

Wildcard principal in ElasticSearch access policy

HighMediumLowNone

SNYK-CC-TF-64

Open this link in a new tab

Encryption at rest is disabled

HighMediumLowNone

SNYK-CC-TF-65

Open this link in a new tab

Encryption at rest is not supported by instance type

HighMediumLowNone

SNYK-CC-TF-66

Open this link in a new tab

Node-to-node encryption is disabled

HighMediumLowNone

SNYK-CC-TF-67

Open this link in a new tab

Cluster does not enforce HTTPS

HighMediumLowNone

SNYK-CC-TF-68

Open this link in a new tab

Cluster does not enforce latest TLS/SSL policy

HighMediumLowNone

SNYK-CC-TF-140

Open this link in a new tab

Elasticsearch domain logging disabled

HighMediumLowNone

Global Accelerator

SNYK-CC-TF-137

Open this link in a new tab

Global Accelerator flow logs disabled

HighMediumLowNone

Glue

SNYK-CC-AWS-444

Open this link in a new tab

Glue policy has wildcard action

HighMediumLowNone

SNYK-CC-AWS-445

Open this link in a new tab

Glue policy has wildcard principal

HighMediumLowNone

SNYK-CC-AWS-448

Open this link in a new tab

Glue CloudWatch log encryption disabled

HighMediumLowNone

SNYK-CC-AWS-449

Open this link in a new tab

Glue job bookmarks encryption disabled

HighMediumLowNone

SNYK-CC-AWS-454

Open this link in a new tab

Glue connection password encryption disabled

HighMediumLowNone

SNYK-CC-AWS-457

Open this link in a new tab

Glue S3 bucket encryption disabled

HighMediumLowNone

IAM

SNYK-CC-AWS-443

Open this link in a new tab

Password reset not required in IAM login profile

HighMediumLowNone

SNYK-CC-AWS-447
Open this link in a new tab

IAM access key generated for `root` user

HighMediumLowNone

SNYK-CC-TF-7
Open this link in a new tab

IAM password should contain uppercase

HighMediumLowNone

SNYK-CC-TF-8
Open this link in a new tab

IAM password should contain lowercase

HighMediumLowNone

SNYK-CC-TF-9
Open this link in a new tab

IAM password should contain symbols

HighMediumLowNone

SNYK-CC-TF-10
Open this link in a new tab

IAM password should contain numbers

HighMediumLowNone

SNYK-CC-TF-11
Open this link in a new tab

IAM requires minimum password length

HighMediumLowNone

SNYK-CC-TF-12
Open this link in a new tab

IAM password reuse prevention is missing

HighMediumLowNone

SNYK-CC-TF-13
Open this link in a new tab

IAM should have max password age

HighMediumLowNone

SNYK-CC-TF-69
Open this link in a new tab
Wildcard action in IAM Policy
HighMediumLowNone

SNYK-CC-TF-116
Open this link in a new tab
IAM Policy attached to user
HighMediumLowNone

SNYK-CC-TF-117
Open this link in a new tab
IAM Role can be assumed by anyone
HighMediumLowNone

SNYK-CC-TF-118
Open this link in a new tab
IAM Role can be assumed by anyone in the account
HighMediumLowNone

SNYK-CC-TF-119
Open this link in a new tab
IAM Policy grants full administrative rights
HighMediumLowNone

Kinesis
SNYK-CC-AWS-450
Open this link in a new tab
Kinesis data stream is not encrypted at rest
HighMediumLowNone

SNYK-CC-TF-62
Open this link in a new tab
Non-Encrypted Kinesis Stream
HighMediumLowNone

KMS

SNYK-CC-TF-14

Open this link in a new tab

KMS key does not have key rotation enabled

HighMediumLowNone

Lambda

SNYK-CC-AWS-436

Open this link in a new tab

Wildcard action in Lambda permission

HighMediumLowNone

SNYK-CC-AWS-437

Open this link in a new tab

Wildcard principal in Lambda permission

HighMediumLowNone

SNYK-CC-TF-122

Open this link in a new tab

Potentially sensitive variable in lambda environment

HighMediumLowNone

SNYK-CC-TF-133

Open this link in a new tab

X-ray tracing is disabled for Lambda function

HighMediumLowNone

Managed Streaming for Kafka (MSK)

SNYK-CC-TF-59

Open this link in a new tab

MSK allows client plaintext traffic

HighMediumLowNone

SNYK-CC-TF-60

Open this link in a new tab

MSK allows in cluster plaintext traffic

HighMediumLowNone

SNYK-CC-TF-139

Open this link in a new tab

MSK Cluster logging disabled

HighMediumLowNone

MQ

SNYK-CC-TF-132

Open this link in a new tab

MQ broker general logs are disabled

HighMediumLowNone

Neptune

SNYK-CC-TF-205

Open this link in a new tab

Non-encrypted Neptune cluster at rest

HighMediumLowNone

Provider

SNYK-CC-TF-74

Open this link in a new tab

Credentials are configured via provider attributes

HighMediumLowNone

Public Access

SNYK-CC-TF-50

Open this link in a new tab

Resource is publicly accessible

HighMediumLowNone

RDS

SNYK-CC-AWS-408

Open this link in a new tab

RDS automatic backup is disabled

HighMediumLowNone

SNYK-CC-AWS-410

Open this link in a new tab

Athena workgroup result encryption is not enforced

HighMediumLowNone

SNYK-CC-AWS-414
Open this link in a new tab
RDS IAM authentication is disabled
HighMediumLowNone

SNYK-CC-AWS-438
Open this link in a new tab
RDS global cluster does not have encryption enabled
HighMediumLowNone

SNYK-CC-AWS-439
Open this link in a new tab
RDS cluster does not have encryption enabled
HighMediumLowNone

SNYK-CC-TF-39
Open this link in a new tab
AWS DB Security Group allows public access
HighMediumLowNone

SNYK-CC-TF-201
Open this link in a new tab
Non-encrypted RDS instance at rest
HighMediumLowNone

Redshift
SNYK-CC-AWS-440
Open this link in a new tab
Redshift cluster does not require SSL connections
HighMediumLowNone

SNYK-CC-TF-38
Open this link in a new tab
AWS Redshift Security Group allows public access
HighMediumLowNone

SNYK-CC-TF-108

Open this link in a new tab

Non-encrypted Redshift DB at rest

HighMediumLowNone

SNYK-CC-TF-136

Open this link in a new tab

Redshift cluster logging disabled

HighMediumLowNone

S3

SNYK-CC-AWS-409

Open this link in a new tab

S3 object is not encrypted

HighMediumLowNone

SNYK-CC-TF-4

Open this link in a new tab

Non-encrypted S3 Bucket

HighMediumLowNone

SNYK-CC-TF-18

Open this link in a new tab

S3 Bucket is publicly readable

HighMediumLowNone

SNYK-CC-TF-19

Open this link in a new tab

S3 Bucket is publicly readable and writable

HighMediumLowNone

SNYK-CC-TF-45

Open this link in a new tab

S3 server access logging is disabled

HighMediumLowNone

SNYK-CC-TF-95

Open this link in a new tab

S3 block public ACLs control is disabled

HighMediumLowNone

SNYK-CC-TF-96

Open this link in a new tab

S3 block public policy control is disabled

HighMediumLowNone

SNYK-CC-TF-97

Open this link in a new tab

S3 ignore public ACLs control is disabled

HighMediumLowNone

SNYK-CC-TF-98

Open this link in a new tab

S3 restrict public bucket control is disabled

HighMediumLowNone

SNYK-CC-TF-121

Open this link in a new tab

S3 Policy grants full rights to anyone

HighMediumLowNone

SNYK-CC-TF-124

Open this link in a new tab

S3 bucket versioning disabled

HighMediumLowNone

SNYK-CC-TF-127

Open this link in a new tab

S3 bucket MFA delete control disabled

HighMediumLowNone

Sagemaker

SNYK-CC-AWS-420

Open this link in a new tab

Sagemaker is not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-AWS-441

Open this link in a new tab

Sagemaker is not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-AWS-442

Open this link in a new tab

Sagemaker data capture location is not encrypted with customer managed key

HighMediumLowNone

Secrets Manager

SNYK-CC-AWS-421

Open this link in a new tab

Secrets Manager is not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-AWS-433

Open this link in a new tab

Wildcard principal in SecretsManager access policy

HighMediumLowNone

SNYK-CC-AWS-434

Open this link in a new tab

Wildcard action in SecretsManager access policy

HighMediumLowNone

SNS

SNYK-CC-AWS-422

Open this link in a new tab

SNS topic is not encrypted with customer managed key

HighMediumLowNone

SNYK-CC-TF-55

Open this link in a new tab

Non-Encrypted SNS Topic

HighMediumLowNone

SQS

SNYK-CC-TF-54

Open this link in a new tab

Non-Encrypted SQS Queue

HighMediumLowNone

SNYK-CC-TF-70

Open this link in a new tab

Wildcard action in SQS Policy

HighMediumLowNone

SSM

SNYK-CC-AWS-456

Open this link in a new tab

SSM session does not use KMS encryption

HighMediumLowNone

SNYK-CC-AWS-458

Open this link in a new tab

SSM S3 data storage not encrypted

HighMediumLowNone

VPC

SNYK-CC-AWS-423

Open this link in a new tab

Default VPC resource detected

HighMediumLowNone

SNYK-CC-AWS-427

Open this link in a new tab

Public IPs are automatically mapped to instances

HighMediumLowNone

SNYK-CC-AWS-428

Open this link in a new tab

Access policy is not attached to the endpoint

HighMediumLowNone

SNYK-CC-TF-1

Open this link in a new tab

Security Group allows open ingress

HighMediumLowNone

SNYK-CC-TF-5

Open this link in a new tab

Default VPC Security Group allows open ingress

HighMediumLowNone

SNYK-CC-TF-6

Open this link in a new tab

Default VPC Security Group allows open egress

HighMediumLowNone

SNYK-CC-TF-37

Open this link in a new tab

AWS Security Group Rule allows public access

HighMediumLowNone

SNYK-CC-TF-40

Open this link in a new tab

AWS Default Network ACL allows public access

HighMediumLowNone

SNYK-CC-TF-41

Open this link in a new tab

AWS Network ACL allows public access

HighMediumLowNone

SNYK-CC-TF-42

Open this link in a new tab

AWS Network ACL Rule allows public access

HighMediumLowNone

SNYK-CC-TF-56

Open this link in a new tab

Missing description

HighMediumLowNone

SNYK-CC-TF-72
Open this link in a new tab
Rule allows open egress
HighMediumLowNone

SNYK-CC-TF-73
Open this link in a new tab
AWS Security Group allows open egress
HighMediumLowNone

WorkSpace
SNYK-CC-AWS-411
Open this link in a new tab
WorkSpace root device encryption is disabled
HighMediumLowNone

SNYK-CC-AWS-412
Open this link in a new tab
WorkSpace user volume encryption is disabled

TYPE
- Terraform

ARM

API Management
SNYK-CC-AZURE-504
Open this link in a new tab
API Management allows anonymous access to developer portal

HighMediumLowNone

SNYK-CC-AZURE-505

Open this link in a new tab

API Management allows insecure TLS/SSL protocols

HighMediumLowNone

SNYK-CC-AZURE-519

Open this link in a new tab

API Management frontend allows insecure TLS/SSL protocols

HighMediumLowNone

App Service (Web Apps)

SNYK-CC-AZURE-500

Open this link in a new tab

Function App does not enforce HTTPS

HighMediumLowNone

SNYK-CC-AZURE-501

Open this link in a new tab

Function App built-in authentication disabled

HighMediumLowNone

SNYK-CC-AZURE-502

Open this link in a new tab

Function App logging disabled

HighMediumLowNone

SNYK-CC-AZURE-533

Open this link in a new tab

App Service allows FTP deployments

HighMediumLowNone

SNYK-CC-AZURE-608

Open this link in a new tab

An SLA should be used for App Configuration

HighMediumLowNone

SNYK-CC-AZURE-618

Open this link in a new tab

Use two or more App Service Plan instances

HighMediumLowNone

SNYK-CC-AZURE-619

Open this link in a new tab

App Service remote debugging enabled

HighMediumLowNone

SNYK-CC-TF-144

Open this link in a new tab

Azure App Service allows HTTP traffic

HighMediumLowNone

SNYK-CC-TF-145

Open this link in a new tab

Azure APP Service does not enforce latest TLS

HighMediumLowNone

SNYK-CC-TF-160

Open this link in a new tab

App Service authentication disabled

HighMediumLowNone

SNYK-CC-TF-161

Open this link in a new tab

App Service identity missing

HighMediumLowNone

SNYK-CC-TF-162

Open this link in a new tab

App Service mutual TLS disabled

HighMediumLowNone

SNYK-CC-TF-163

Open this link in a new tab

App Service HTTP/2 disabled

HighMediumLowNone

SNYK-CC-TF-245
Open this link in a new tab
App Service not running latest .Net version
HighMediumLowNone

SNYK-CC-TF-246
Open this link in a new tab
App Service not running latest PHP version
HighMediumLowNone

SNYK-CC-TF-247
Open this link in a new tab
App Service not running latest Python version
HighMediumLowNone

SNYK-CC-TF-248
Open this link in a new tab
App Service not running latest Java version
HighMediumLowNone

Authorization
SNYK-CC-TF-150
Open this link in a new tab
Custom subscription owner role
HighMediumLowNone

Automation
SNYK-CC-AZURE-620
Open this link in a new tab
Ensure Automation Variables are encrypted
HighMediumLowNone

Batch
SNYK-CC-AZURE-508
Open this link in a new tab
Batch job runs in admin mode

HighMediumLowNone

CDN

SNYK-CC-AZURE-509

Open this link in a new tab

CDN Endpoint https not enforced

HighMediumLowNone

Compute

SNYK-CC-AZURE-475

Open this link in a new tab

Linux VM scale set encryption at host disabled

HighMediumLowNone

SNYK-CC-AZURE-478

Open this link in a new tab

Windows VM scale set encryption at host disabled

HighMediumLowNone

SNYK-CC-TF-77

Open this link in a new tab

Encryption at rest is disabled

HighMediumLowNone

SNYK-CC-TF-79

Open this link in a new tab

Linux virtual machine has password authentication enabled

HighMediumLowNone

SNYK-CC-TF-182

Open this link in a new tab

Virtual machine configuration contains sensitive data

HighMediumLowNone

SNYK-CC-TF-263

Open this link in a new tab

Hardcoded admin password in VM configuration

HighMediumLowNone

Container

SNYK-CC-AZURE-594

Open this link in a new tab

Azure Container Registry Admin is enabled

HighMediumLowNone

SNYK-CC-AZURE-597

Open this link in a new tab

Container image quarantine disabled

HighMediumLowNone

SNYK-CC-TF-80

Open this link in a new tab

RBAC is disabled

HighMediumLowNone

SNYK-CC-TF-81

Open this link in a new tab

API Server allows public access

HighMediumLowNone

SNYK-CC-TF-82

Open this link in a new tab

Container Insights is disabled for AKS

HighMediumLowNone

SNYK-CC-TF-176

Open this link in a new tab

AKS Network Policies disabled

HighMediumLowNone

SNYK-CC-TF-177

Open this link in a new tab

AKS Kubernetes Dashboard enabled

HighMediumLowNone

CosmosDB (DocumentDB)

SNYK-CC-AZURE-510

CosmosDB account automatic failover disabled

HighMediumLowNone

SNYK-CC-AZURE-511

CosmosDB account public network access enabled

HighMediumLowNone

SNYK-CC-AZURE-512

CosmosDB account acl bypass for trusted services enabled

HighMediumLowNone

SNYK-CC-AZURE-621

Restrict user access to data operations in Azure Cosmos DB

HighMediumLowNone

Data Factory

SNYK-CC-AZURE-513

Data Factory public access enabled

HighMediumLowNone

SNYK-CC-AZURE-514

Data Factory not encrypted with customer managed key

HighMediumLowNone

Data Lake

SNYK-CC-AZURE-515

Data Lake Storage firewall disabled

HighMediumLowNone

SNYK-CC-TF-23

Azure Data Lake Analytics Firewall Rule allows public access
HighMediumLowNone

SNYK-CC-TF-24
Azure Data Lake Store Firewall Rule allows public access
HighMediumLowNone

SNYK-CC-TF-78
Non-Encrypted Data Lake Store
HighMediumLowNone

Database
SNYK-CC-AZURE-468
Azure SQL database audit retention period is below 90 days
HighMediumLowNone

SNYK-CC-AZURE-470
MariaDB geo-redundant backup disabled
HighMediumLowNone

SNYK-CC-AZURE-471
MariaDB public access is enabled
HighMediumLowNone

SNYK-CC-AZURE-472
MariaDB server does not enforce SSL
HighMediumLowNone

SNYK-CC-AZURE-539
Periodic vulnerability assessment is not enabled on SQL server

HighMediumLowNone

SNYK-CC-AZURE-540

Open this link in a new tab

MSSQL Vulnerability Assessment email notifications are disabled

HighMediumLowNone

SNYK-CC-TF-25

Open this link in a new tab

Azure MariaDB Firewall Rule allows public access

HighMediumLowNone

SNYK-CC-TF-26

Open this link in a new tab

Azure Mysql Firewall Rule allows public access

HighMediumLowNone

SNYK-CC-TF-27

Open this link in a new tab

Azure Postgresql Firewall Rule allows public access

HighMediumLowNone

SNYK-CC-TF-29

Open this link in a new tab

Azure SQL Firewall Rule allows public access

HighMediumLowNone

SNYK-CC-TF-146

Open this link in a new tab

Azure MySQL Server does not enforce SSL

HighMediumLowNone

SNYK-CC-TF-147

Open this link in a new tab

Azure Postgresql Server does not enforce SSL

HighMediumLowNone

SNYK-CC-TF-153

Azure SQL server audit retention period is below 90 days

HighMediumLowNone

SNYK-CC-TF-154

Azure PostgreSQL 'log_checkpoint' is disabled

HighMediumLowNone

SNYK-CC-TF-167

SQL Server auditing is disabled

HighMediumLowNone

SNYK-CC-TF-168

SQL Advanced Threat detection type is disabled

HighMediumLowNone

SNYK-CC-TF-169

Missing email recipients for SQL threat alerts

HighMediumLowNone

SNYK-CC-TF-170

SQL Advanced Threat alerts are not send to admins

HighMediumLowNone

SNYK-CC-TF-171

The connection_throttling setting is disabled on PostgreSQL DB

HighMediumLowNone

SNYK-CC-TF-257

The log_connections setting is disabled on Postgresql Server

HighMediumLowNone

SNYK-CC-TF-258

Open this link in a new tab

Azure Postgresql server log_disconnects setting is disabled

HighMediumLowNone

SNYK-CC-TF-259

Open this link in a new tab

The log_duration setting is not configured on Postgresql server

HighMediumLowNone

SNYK-CC-TF-260

Open this link in a new tab

Azure Postgresql log_retention_days setting is below 3 days

HighMediumLowNone

Key Vault

SNYK-CC-AZURE-625

Open this link in a new tab

Key Vault has soft deletion disabled

HighMediumLowNone

SNYK-CC-TF-173

Open this link in a new tab

Vault key expiration date not set

HighMediumLowNone

SNYK-CC-TF-174

Open this link in a new tab

Vault secret expiration date not set

HighMediumLowNone

SNYK-CC-TF-175

Open this link in a new tab

Vault accidental key deletion prevention disabled

HighMediumLowNone

Monitor

SNYK-CC-AZURE-552

Open this link in a new tab

Ensure Diagnostic Setting captures appropriate categories

HighMediumLowNone

SNYK-CC-TF-156

Open this link in a new tab

Insufficient Activity Log retention period

HighMediumLowNone

SNYK-CC-TF-157

Open this link in a new tab

Audit profile does not capture all categories

HighMediumLowNone

Network

SNYK-CC-AZURE-474

Open this link in a new tab

WAF not enabled on application gateway

HighMediumLowNone

SNYK-CC-AZURE-476

Open this link in a new tab

FrontDoor WAF disabled

HighMediumLowNone

SNYK-CC-AZURE-516

Open this link in a new tab

Virtual Network DDoS protection plan disabled

HighMediumLowNone

SNYK-CC-AZURE-610

Open this link in a new tab

App Gateway does not use OWASP 3.x rules

HighMediumLowNone

SNYK-CC-AZURE-611

Open this link in a new tab

WAF prevention mode not enabled

HighMediumLowNone

SNYK-CC-TF-20
Open this link in a new tab
Azure Firewall Network Rule Collection allows public access

HighMediumLowNone

SNYK-CC-TF-21
Open this link in a new tab
Azure Firewall Application Rule allows public access

HighMediumLowNone

SNYK-CC-TF-33
Open this link in a new tab
Azure Network Security Group allows public access

HighMediumLowNone

SNYK-CC-TF-35
Open this link in a new tab
Azure Network Security Rule allows public access

HighMediumLowNone

SNYK-CC-TF-76
Open this link in a new tab
Azure Network Security Rule allows open egress

HighMediumLowNone

SNYK-CC-TF-152
Open this link in a new tab
NSG Flow Log retention period is below 90 days

HighMediumLowNone

Redis
SNYK-CC-AZURE-477
Open this link in a new tab
Redis cache insecure port enabled

HighMediumLowNone

SNYK-CC-AZURE-517

Open this link in a new tab

Redis Cache accessible without authentication

HighMediumLowNone

SNYK-CC-AZURE-518

Open this link in a new tab

Redis Cache backup disabled

HighMediumLowNone

SNYK-CC-TF-28

Open this link in a new tab

Azure Redis Firewall Rule allows public access

HighMediumLowNone

Security Center

SNYK-CC-AZURE-521

Open this link in a new tab

Azure Defender is disabled on SQL servers

HighMediumLowNone

SNYK-CC-AZURE-522

Open this link in a new tab

App Service is not protected by Azure Defender

HighMediumLowNone

SNYK-CC-AZURE-524

Open this link in a new tab

Azure Defender is disabled on SQL server virtual machines

HighMediumLowNone

SNYK-CC-AZURE-525

Open this link in a new tab

Azure Defender is disabled on storage accounts

HighMediumLowNone

SNYK-CC-AZURE-526

Azure Defender is disabled on kubernetes services

HighMediumLowNone

SNYK-CC-AZURE-527

Azure Defender is disabled on container registries

HighMediumLowNone

SNYK-CC-AZURE-528

Key Vault is not protected by Azure Defender

HighMediumLowNone

SNYK-CC-AZURE-529

Automatic provisioning of monitoring agent is disabled

HighMediumLowNone

SNYK-CC-AZURE-532

High severity security notifications disabled

HighMediumLowNone

SNYK-CC-AZURE-537

Advanced Threat Protection is disabled on SQL server

HighMediumLowNone

SNYK-CC-AZURE-542

Windows Defender ATP (WDATP) integration in Security Center disabled

HighMediumLowNone

SNYK-CC-AZURE-543

Microsoft Cloud App Security (MCAS) integration in Security Center disabled

HighMediumLowNone

SNYK-CC-AZURE-553

Open this link in a new tab

Azure Defender is disabled for Virtual Machines

HighMediumLowNone

SNYK-CC-AZURE-554

Open this link in a new tab

Azure Defender is disabled for Azure Resource Management (ARM)

HighMediumLowNone

SNYK-CC-AZURE-555

Open this link in a new tab

Azure Defender is disabled for Azure DNS

HighMediumLowNone

SNYK-CC-AZURE-624

Open this link in a new tab

Key Vault purge protection is disabled

HighMediumLowNone

SNYK-CC-TF-164

Open this link in a new tab

Security Center Free Tier

HighMediumLowNone

SNYK-CC-TF-165

Open this link in a new tab

Security Center missing phone number

HighMediumLowNone

SNYK-CC-TF-166

Open this link in a new tab

Security Center alert notification disabled

HighMediumLowNone

Service Fabric

SNYK-CC-AZURE-473

Open this link in a new tab

Service fabric does not use active directory authentication

HighMediumLowNone

Storage

SNYK-CC-AZURE-534

Open this link in a new tab

Storage Queue service logging is disabled

HighMediumLowNone

SNYK-CC-AZURE-535

Open this link in a new tab

Public access level for storage containers & blobs is enabled

HighMediumLowNone

SNYK-CC-TF-148

Open this link in a new tab

Storage Account does not enforce HTTPS

HighMediumLowNone

SNYK-CC-TF-149

Open this link in a new tab

Storage Account does not enforce latest TLS

HighMediumLowNone

SNYK-CC-TF-172

Open this link in a new tab

Trusted Microsoft Service access to storage account is disabled

HighMediumLowNone

SNYK-CC-TF-179

Open this link in a new tab

Storage Account allows any traffic by default

HighMediumLowNone

SNYK-CC-TF-181

Open this link in a new tab

Storage container allows public access

HighMediumLowNone

SNYK-CC-TF-244

Open this link in a new tab

SAS token can be used over insecure HTTP

HighMediumLowNone

Synapse

SNYK-CC-TF-30

Open this link in a new tab

Azure Synapse Firewall Rule allows public access

TYPE

- Terraform

App Engine

SNYK-CC-GCP-297

Open this link in a new tab

App Engine application does not enforce HTTPS

HighMediumLowNone

SNYK-CC-TF-31

Open this link in a new tab

GCP App Engine Firewall Rule allows public access

HighMediumLowNone

BigQuery

SNYK-CC-TF-236

Open this link in a new tab

BigQuery dataset is publicly accessible

HighMediumLowNone

Cloud DNS

SNYK-CC-GCP-273

Open this link in a new tab

DNSSEC is not enabled on managed zone

HighMediumLowNone

SNYK-CC-GCP-276

Open this link in a new tab

DNSSEC is configured with weak algorithm

HighMediumLowNone

Cloud Key Management Service

SNYK-CC-GCP-278

Open this link in a new tab

KMS keys are not automatically rotated

HighMediumLowNone

SNYK-CC-GCP-301

Open this link in a new tab

KMS key openly accessible

HighMediumLowNone

Cloud Platform

SNYK-CC-GCP-272

Open this link in a new tab

IAM audit logging configuration does not encompass all events and users

HighMediumLowNone

SNYK-CC-GCP-275

Open this link in a new tab

Default network automatically created

HighMediumLowNone

Cloud SQL

SNYK-CC-GCP-270

Open this link in a new tab

SSL is not enabled on CloudSQL instance

HighMediumLowNone

SNYK-CC-GCP-283

Open this link in a new tab

Cloud SQL instance backup disabled

HighMediumLowNone

SNYK-CC-GCP-287

Open this link in a new tab

The log_checkpoints is disabled on PostgreSQL DB

HighMediumLowNone

SNYK-CC-GCP-288

Open this link in a new tab

The log_connections setting is disabled on Postgresql DB

HighMediumLowNone

The log_disconnections setting is disabled on PostgreSQL DB
HighMediumLowNone

The log_lock_waits setting is disabled on PostgreSQL DB
HighMediumLowNone

Temporary file information is not logged
HighMediumLowNone

SQL statements may be logged
HighMediumLowNone

Cross DB ownership chaining is enabled
HighMediumLowNone

Contained database authentication is enabled
HighMediumLowNone

MySQL allows loading local data
HighMediumLowNone

Cloud SQL instance is publicly accessible
HighMediumLowNone

Public IP assigned to SQL database instance
HighMediumLowNone

Cloud Storage

SNYK-CC-GCP-271

Open this link in a new tab

Object versioning is not enabled

HighMediumLowNone

SNYK-CC-GCP-274

Open this link in a new tab

Logging is not enabled on storage bucket

HighMediumLowNone

SNYK-CC-TF-185

Open this link in a new tab

CMKs are not used to encrypt storage bucket

HighMediumLowNone

SNYK-CC-TF-239

Open this link in a new tab

Cloud Storage bucket is publicly accessible

HighMediumLowNone

SNYK-CC-TF-240

Open this link in a new tab

Uniform bucket-level access disabled

HighMediumLowNone

Compute Engine

SNYK-CC-GCP-277

Open this link in a new tab

IP forwarding is enabled

HighMediumLowNone

SNYK-CC-GCP-279

Open this link in a new tab

OS Login disabled

HighMediumLowNone

SNYK-CC-GCP-280

Open this link in a new tab

OS Login disabled on instance

HighMediumLowNone

SNYK-CC-GCP-281

Open this link in a new tab

Project wide ssh keys are allowed

HighMediumLowNone

Shielded VM disabled
HighMediumLowNone

Default service account used
HighMediumLowNone

Serial port enabled
HighMediumLowNone

SSL policy allows weak algorithms
HighMediumLowNone

Unrestricted SSH access
HighMediumLowNone

Unrestricted RDP access
HighMediumLowNone

Backend service is not enforcing HTTPS
HighMediumLowNone

Backend service logging is disabled
HighMediumLowNone

Delete protection is disabled
HighMediumLowNone

IP forwarding enabled on the instance
HighMediumLowNone

Encryption key exposed in instance template configuration
HighMediumLowNone

IP forwarding enabled on the instance template
HighMediumLowNone

Packet mirroring resource in use
HighMediumLowNone

Sensitive certificate key material is stored in state file
HighMediumLowNone

GCP Compute Firewall allows public access
HighMediumLowNone

GCP Compute Firewall allows open egress
HighMediumLowNone

Customer supplied encryption keys are not used to encrypt compute disk
HighMediumLowNone

Customer supplied encryption keys are not used to encrypt VM compute instance
HighMediumLowNone

SNYK-CC-TF-194

VPC Flow Logs for VPC Network Subnets Disabled

HighMediumLowNone

SNYK-CC-TF-241

Public IP assigned to compute instance

HighMediumLowNone

IAM

SNYK-CC-GCP-298

IAM user with privileged roles at project level

HighMediumLowNone

SNYK-CC-GCP-299

Service account with admin privileges

HighMediumLowNone

SNYK-CC-TF-90

IAM permission assigned to user

HighMediumLowNone

Kubernetes (Container) Engine

SNYK-CC-TF-83

Legacy ABAC is enabled

HighMediumLowNone

SNYK-CC-TF-84

GKE Workload Identity is disabled

HighMediumLowNone

SNYK-CC-TF-85

GKE legacy endpoint enabled

HighMediumLowNone

SNYK-CC-TF-86

GKE Basic Auth is enabled

HighMediumLowNone

SNYK-CC-TF-87
Open this link in a new tab
GKE client certificate auth is enabled
HighMediumLowNone

SNYK-CC-TF-88
Open this link in a new tab
GKE PodSecurityPolicy controller is disabled
HighMediumLowNone

SNYK-CC-TF-89
Open this link in a new tab
GKE Shield is disabled
HighMediumLowNone

SNYK-CC-TF-193
Open this link in a new tab
Logging is disabled on Kubernetes Engine Clusters
HighMediumLowNone

SNYK-CC-TF-195
Open this link in a new tab
GKE Node Pool auto upgrade disabled
HighMediumLowNone

SNYK-CC-TF-196
Open this link in a new tab
GKE Network Policies enforcement disabled
HighMediumLowNone

SNYK-CC-TF-202
Open this link in a new tab
GKE cluster labels are missing
HighMediumLowNone

SNYK-CC-TF-203
Open this link in a new tab
GKE Node pool does not use COS
HighMediumLowNone

SNYK-CC-TF-231
Open this link in a new tab
GKE Stackdriver monitoring disabled
HighMediumLowNone

SNYK-CC-TF-232

Open this link in a new tab

GKE Node Pool auto repair disabled

HighMediumLowNone

SNYK-CC-TF-237

Open this link in a new tab

GKE control plane is publicly accessible

HighMediumLowNone

SNYK-CC-TF-238

Open this link in a new tab

GKE private cluster options is not enabled

HighMediumLowNone

SNYK-CC-TF-269

Open this link in a new tab

GKE Alias IP disabled

TYPE

- Kubernetes

Deployment

SNYK-CC-K8S-1

Open this link in a new tab

Container is running in privileged mode

HighMediumLowNone

SNYK-CC-K8S-2

Open this link in a new tab

Container is running with docker socket mount

HighMediumLowNone

SNYK-CC-K8S-3

Open this link in a new tab

Container is running in host's PID namespace

HighMediumLowNone

SNYK-CC-K8S-4

Open this link in a new tab

Container is running without memory limit

HighMediumLowNone

SNYK-CC-K8S-5

Open this link in a new tab

Container is running without cpu limit

HighMediumLowNone

SNYK-CC-K8S-6

Open this link in a new tab

Container does not drop all default capabilities

HighMediumLowNone

SNYK-CC-K8S-7

Open this link in a new tab

Container is running with SYS_ADMIN capability

HighMediumLowNone

SNYK-CC-K8S-8

Open this link in a new tab

Container is running with writable root filesystem

HighMediumLowNone

SNYK-CC-K8S-9

Open this link in a new tab

Container is running without privilege escalation control

HighMediumLowNone

SNYK-CC-K8S-10

Open this link in a new tab

Container is running without root user control

HighMediumLowNone

SNYK-CC-K8S-11

Container's UID could clash with host's UID

HighMediumLowNone

SNYK-CC-K8S-12

Container is running with custom hosts file configuration

HighMediumLowNone

SNYK-CC-K8S-13

Container is running in host's IPC namespace

HighMediumLowNone

SNYK-CC-K8S-14

Container is running in host's Network namespace

HighMediumLowNone

SNYK-CC-K8S-32

Container is running without AppArmor profile

HighMediumLowNone

SNYK-CC-K8S-33

Container is running with custom SELinux options

HighMediumLowNone

SNYK-CC-K8S-34

Container is exposing SSH port

HighMediumLowNone

SNYK-CC-K8S-36

Container is running with multiple open ports

HighMediumLowNone

SNYK-CC-K8S-37

Open this link in a new tab

Container is running with host device path mount

HighMediumLowNone

SNYK-CC-K8S-38

Open this link in a new tab

Container is running with shared mount propagation

HighMediumLowNone

SNYK-CC-K8S-41

Open this link in a new tab

Container is running without liveness probe

HighMediumLowNone

SNYK-CC-K8S-42

Open this link in a new tab

Container could be running with outdated image

HighMediumLowNone

Network Policy

SNYK-CC-K8S-25

Open this link in a new tab

Policy does not restrict ingress sources

HighMediumLowNone

SNYK-CC-K8S-26

Open this link in a new tab

Policy does not restrict egress destinations

HighMediumLowNone

Pod Security Policy

SNYK-CC-K8S-16

Open this link in a new tab

Policy allows `privileged` containers

HighMediumLowNone

SNYK-CC-K8S-17

Open this link in a new tab

Policy allows privilege escalation

HighMediumLowNone

SNYK-CC-K8S-18

Open this link in a new tab

Policy sets insecure default privilege escalation control

HighMediumLowNone

SNYK-CC-K8S-19

Open this link in a new tab

Policy allows sharing of host's PID namespace

HighMediumLowNone

SNYK-CC-K8S-20

Open this link in a new tab

Policy allows sharing of host's IPC namespace

HighMediumLowNone

SNYK-CC-K8S-21

Open this link in a new tab

Policy allows sharing of host's Network namespace

HighMediumLowNone

SNYK-CC-K8S-22

Open this link in a new tab

Policy does not prevent use of root user

HighMediumLowNone

SNYK-CC-K8S-23

Open this link in a new tab

Policy allows all capabilities

HighMediumLowNone

SNYK-CC-K8S-24

Open this link in a new tab

Policy does not restrict default capabilities

HighMediumLowNone

SNYK-CC-K8S-27
Open this link in a new tab
Policy does not prevent use of root group
HighMediumLowNone

SNYK-CC-K8S-28
Open this link in a new tab
Policy allows insecure seccomp profiles
HighMediumLowNone

SNYK-CC-K8S-29
Open this link in a new tab
Policy sets insecure default seccomp profile
HighMediumLowNone

SNYK-CC-K8S-30
Open this link in a new tab
Policy allows all volume types
HighMediumLowNone

SNYK-CC-K8S-31
Open this link in a new tab
Policy allows any apparmor profile
HighMediumLowNone

Role
SNYK-CC-K8S-44
Open this link in a new tab
Role with too wide permissions
HighMediumLowNone

RoleBinding
SNYK-CC-K8S-45
Open this link in a new tab
RoleBinding to the default service account
HighMediumLowNone

SNYK-CC-K8S-46

Open this link in a new tab

Binding using a pre-defined role

HighMediumLowNone

SNYK-CC-K8S-47

Open this link in a new tab

Role with dangerous permissions

HighMediumLowNone

SNYK-CC-K8S-48

Open this link in a new tab

Role that grant permissions to system reserved namespace

HighMediumLowNone

Service

SNYK-CC-K8S-15

Open this link in a new tab

Service does not restrict ingress sources

HighMediumLowNone

SNYK-CC-K8S-43

Open this link in a new tab

Service is using an external IP

CODES

accelerometer, autoplay, camera, ch-device-memory, ch-downlink, ch-dpr, ch-ect, ch-lang, ch-prefers-color-scheme, ch-rtt, ch-ua, ch-ua-arch, ch-ua-bitness, ch-ua-full-version, ch-ua-mobile, ch-ua-model, ch-ua-platform, ch-ua-platform-version, ch-ua-reduced, ch-viewport-width, ch-width, clipboard-read, clipboard-write, cross-origin-isolated, display-capture, document-domain, encrypted-media, fullscreen, geolocation, gyroscope, hid, idle-detection, magnetometer, microphone, midi, otp-credentials, payment, picture-in-picture, publickey-credentials-get, screen-wake-lock, serial, sync-xhr, usb, xr-spatial-tracking
https://accounts.shopify.com/oauth/authorize?
client_id=7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775
c&destination_uuid=fdf6184b-4c7c-4160-

ab1b-7d11f36a8ca8&nonce=229afc745a2f7a3b007d2598b20fe7b8&prompt=merge&redirect_uri=https://ama-sub-provider.myshopify.com/admin/auth/identity/callback&response_type=code&scope=email https://api.shopify.com/auth/destinations.readonly openid profile https://api.shopify.com/auth/partners.collaborator-relationships.readonly https://api.shopify.com/auth/banking.manage https://api.shopify.com/auth/shopify-chat.admin.graphql https://api.shopify.com/auth/flow.workflows.manage https://api.shopify.com/auth/organization-identity.manage&state=8c8edd20a078f9fd8ab527ffc8fb1693&ui_locales=en&ux=shop

———————————————————————

https://accounts.shopify.com/oauth/authorize?client_id=7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775c&destination_uuid=fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8&nonce=229afc745a2f7a3b007d2598b20fe7b8&prompt=merge&redirect_uri=https://ama-sub-provider.myshopify.com/admin/auth/identity/callback&response_type=code&scope=email https://api.shopify.com/auth/destinations.readonly openid profile https://api.shopify.com/auth/partners.collaborator-relationships.readonly https://api.shopify.com/auth/banking.manage https://api.shopify.com/auth/shopify-chat.admin.graphql https://api.shopify.com/auth/flow.workflows.manage https://api.shopify.com/auth/organization-identity.manage&state=8c8edd20a078f9fd8ab527ffc8fb1693&ui_locales=en&ux=shop
client_id=7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775c&destination_uuid=fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8&nonce=229afc745a2f7a3b007d2598b20fe7b8&prompt=merge&redirect_uri=https%3A%2F%2Fama-sub-provider.myshopify.com%2Fadmin%2Fauth%2Fidentity%2Fcallback&response_type=code&scope=email%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Fdestinations.readonly%20openid%20profile%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Fpartners.collaborator-relationships.readonly%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Fbanking.manage%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Fshopify-chat.admin.graphql%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Fflow.workflows.manage%20https%3A%2F%2Fapi.shopify.com%2Fauth%2Forganization-identity.manage&state=8c8edd20a078f9fd8ab527ffc8fb1693&ui_locales=en&ux=shop

———————————————————————

https://ama-sub-provider.myshopify.com/
"Chromium";v="94", "Google Chrome";v="94", ";Not A Brand";v="99"
Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/94.0.4606.61 Mobile Safari/537.36
7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775c
fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8 229afc745a2f7a3b007d2598b20fe7b8
229afc745a2f7a3b007d2598b20fe7b8 https://ama-sub-provider.myshopify.com/admin/auth/identity/callback
code
email https://api.shopify.com/auth/destinations.readonly openid profile https://api.shopify.com/auth/partners.collaborator-relationships.readonly https://api.shopify.com/auth/banking.manage https://api.shopify.com/auth/shopify-chat.admin.graphql https://api.shopify.com/auth/flow.workflows.manage https://

api.shopify.com/auth/organization-identity.manage
8c8edd20a078f9fd8ab527ffc8fb1693 en
shop
————————————————————————
7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775c
fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8 229afc745a2f7a3b007d2598b20fe7b8
merge
https://ama-sub-provider.myshopify.com/admin/auth/identity/callback
code
email https://api.shopify.com/auth/destinations.readonly openid profile https://
api.shopify.com/auth/partners.collaborator-relationships.readonly https://
api.shopify.com/auth/banking.manage https://api.shopify.com/auth/shopify-
chat.admin.graphql https://api.shopify.com/auth/flow.workflows.manage https://
api.shopify.com/auth/organization-identity.manage
8c8edd20a078f9fd8ab527ffc8fb1693
en
shop
————————————————————————— https://ama-sub-
provider.myshopify.com/
"Chromium";v="94", "Google Chrome";v="94", ";Not A Brand";v="99" ?1
"Android" 1
Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/94.0.4606.61 Mobile Safari/537.36
7ee65a63608843c577db8b23c4d7316ea0a01bd2f7594f8a9c06ea668c1b775c
fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8 https://ama-sub-provider.myshopify.com/
admin/auth/identity/callback
email https://api.shopify.com/auth/destinations.readonly openid profile https://
api.shopify.com/auth/partners.collaborator-relationships.readonly https://
api.shopify.com/auth/banking.manage https://api.shopify.com/auth/shopify-
chat.admin.graphql https://api.shopify.com/auth/flow.workflows.manage https://
api.shopify.com/auth/organization-identity.manage
"Chromium";v="94", "Google Chrome";v="94", ";Not A Brand";v="99"
Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML,
like Gecko) Chrome/94.0.4606.61 Mobile Safari/537.36
fdf6184b-4c7c-4160-ab1b-7d11f36a8ca8
————————————————————————
VM156:1250 crbug/1173575, non-JS module files deprecated. (anonymous) @
VM156:1250

```
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// #import {assertInstanceof} from './assert.m.js'; // #import {dispatchSimpleEvent} from
'./cr.m.js';
/**
* Alias for document.getElementById. Found elements must be HTMLElements. *
@param {string} id The ID of the element to find.
```

```
 * @return {HTMLElement} The found element or null if not found.
 */
/* #export */ function $(id) {
// Disable getElementById restriction here, since we are instructing other // places to re-
use the $() that is defined here.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, HTMLElement) : null;
}
// TODO(devlin): This should return SVGElement, but closure compiler is missing //
those externs.
/**
 * Alias for document.getElementById. Found elements must be SVGElements. *
 @param {string} id The ID of the element to find.
 * @return {Element} The found element or null if not found.
 */
/* #export */ function getSVGElement(id) {
// Disable getElementById restriction here, since it is not suitable for SVG // elements.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, Element) : null;
}
/**
 * @return {?Element} The currently focused element (including elements that are *
 behind a shadow root), or null if nothing is focused.
 */
/* #export */ function getDeepActiveElement() {
let a = document.activeElement;
while (a && a.shadowRoot && a.shadowRoot.activeElement) {
a = a.shadowRoot.activeElement; }
return a; }
//
/**
 * @param {Node} el A node to search for ancestors with |className|.
 * @param {string} className A class to search for.
 * @return {Element} A node with class of |className| or null if none is found. */
/* #export */ function findAncestorByClass(el, className) { return /** @type {Element}
*/ (findAncestor(el, function(el) {
return el.classList && el.classList.contains(className); }));
}
/**
```
-      `* Return the first ancestor for which the {@code predicate} returns true.`
-      `* @param {Node} node The node to check.`
-      `* @param {function(Node):boolean} predicate The function that tests the`

- * nodes.

- * @param {boolean=} includeShadowHosts

- * @return {Node} The found ancestor or null if not found. */
/* #export */ function findAncestor(node, predicate, includeShadowHosts) { while (node !== null) {
if (predicate(node)) { break;
}
node = includeShadowHosts && node instanceof ShadowRoot ? node.host :

}
return node; }
node.parentNode;
/**
* Disables text selection and dragging, with optional callbacks to specify
* overrides.
* @param {function(Event):boolean=} opt_allowSelectStart Unless this function * is defined and returns true, the onselectionstart event will be
* suppressed.
* @param {function(Event):boolean=} opt_allowDragStart Unless this function * is defined and returns true, the ondragstart event will be suppressed.
*/
/* #export */ function disableTextSelectAndDrag(opt_allowSelectStart, opt_allowDragStart) {
// Disable text selection. document.onselectstart = function(e) {
if (!(opt_allowSelectStart && opt_allowSelectStart.call(this, e))) { e.preventDefault();
} };
// Disable dragging. document.ondragstart = function(e) {
if (!(opt_allowDragStart && opt_allowDragStart.call(this, e))) { e.preventDefault();
} };
}
/**
* Check the directionality of the page.
* @return {boolean} True if Chrome is running an RTL UI. */
/* #export */ function isRTL() {
return document.documentElement.dir === 'rtl';
}
/**
* Get an element that's known to exist by its ID. We use this instead of just * calling getElementById and not checking the result because this lets us
* satisfy the JSCompiler type system.
* @param {string} id The identifier name.
* @return {!HTMLElement} the Element.
*/

```
/* #export */ function getRequiredElement(id) { return assertInstanceof(
$(id), HTMLElement, 'Missing required element: ' + id); }
/**
```

- `* Query an element that's known to exist by a selector. We use this instead of`

- `* just calling querySelector and not checking the result because this lets us`

- `* satisfy the JSCompiler type system.`

- `* @param {string} selectors CSS selectors to query the element.`

- `* @param {(!Document|!DocumentFragment|!Element)=} opt_context An optional`

- `* context object for querySelector.`

- `* @return {!HTMLElement} the Element. */`
  ```
  /* #export */ function queryRequiredElement(selectors, opt_context) { const element
  = (opt_context || document).querySelector(selectors); return assertInstanceof(
  element, HTMLElement, 'Missing required element: ' + selectors); }
  /**
  ```

- `* Creates a new URL which is the old URL with a GET param of key=value.`

- `* @param {string} url The base URL. There is not sanity checking on the URL so`

- `* it must be passed in a proper format.`

- `* @param {string} key The key of the param.`

- `* @param {string} value The value of the param.`

- `* @return {string} The new URL. */`
  ```
  /* #export */ function appendParam(url, key, value) {
  const param = encodeURIComponent(key) + '=' + encodeURIComponent(value);
  if (url.indexOf('?') === -1) { return url + '?' + param;
  }
  return url + '&' + param; }
  /**
  * Creates an element of a specified type with a specified class name. * @param
  {string} type The node type.
  * @param {string} className The class name to use.
  ```

```
     * @return {Element} The created element.
     */
    /* #export */ function createElementWithClassName(type, className) { const elm =
    document.createElement(type);
    elm.className = className;
    return elm;
    }
    /**
     * transitionend does not always fire (e.g. when animation is aborted
```

```
* or when no paint happens during the animation). This function sets up
* a timer and emulate the event if it is not fired when the timer expires.
* @param {!HTMLElement} el The element to watch for transitionend.
* @param {number=} opt_timeOut The maximum wait time in milliseconds for the *
transitionend to happen. If not specified, it is fetched from |el|
* using the transitionDuration style value.
*/
/* #export */ function ensureTransitionEndEvent(el, opt_timeOut) {
if (opt_timeOut === undefined) {
const style = getComputedStyle(el);
opt_timeOut = parseFloat(style.transitionDuration) * 1000;
// Give an additional 50ms buffer for the animation to complete.
opt_timeOut += 50; }
let fired = false; el.addEventListener('transitionend', function f(e) {
el.removeEventListener('transitionend', f);
fired = true; });
window.setTimeout(function() { if (!fired) {
cr.dispatchSimpleEvent(el, 'transitionend', true); }
}, opt_timeOut); }
/**
* Alias for document.scrollTop getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The Y document scroll offset.
*/
/* #export */ function scrollTopForDocument(doc) {
return doc.documentElement.scrollTop || doc.body.scrollTop;
}
/**
* Alias for document.scrollTop setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target Y scroll offset.
*/
/* #export */ function setScrollTopForDocument(doc, value)
{ doc.documentElement.scrollTop = doc.body.scrollTop = value;
```

```
}
/**
 * Alias for document.scrollLeft getter.
 * @param {!HTMLDocument} doc The document node where information will be *
 queried from.
 * @return {number} The X document scroll offset.
 */
/* #export */ function scrollLeftForDocument(doc) {
return doc.documentElement.scrollLeft || doc.body.scrollLeft;
}
/**
 * Alias for document.scrollLeft setter.
 * @param {!HTMLDocument} doc The document node where information will be *
 queried from.
 * @param {number} value The target X scroll offset.
 */
/* #export */ function setScrollLeftForDocument(doc, value)
{ doc.documentElement.scrollLeft = doc.body.scrollLeft = value;
}
/**
 * Replaces '&', '<', '>', '"', and ''' characters with their HTML encoding.
 * @param {string} original The original string.
 * @return {string} The string with all the characters mentioned above replaced. */
/* #export */ function HTMLEscape(original) { return original.replace(/&/g, '&amp;')
.replace(/</g, '&lt;') .replace(/>/g, '&gt;') .replace(/"/g, '&quot;') .replace(/'/g, '&#39;');
}
/**
 * Shortens the provided string (if necessary) to a string of length at most
 * |maxLength|.
 * @param {string} original The original string.
 * @param {number} maxLength The maximum length allowed for the string. * @return
 {string} The original string if its length does not exceed
 * |maxLength|. Otherwise the first |maxLength| - 1 characters with '...'
 * appended.
 */
/* #export */ function elide(original, maxLength) { if (original.length <= maxLength) {
return original; }
return original.substring(0, maxLength - 1) + '\u2026'; }
/**
 * Quote a string so it can be used in a regular expression. * @param {string} str The
 source string.
 * @return {string} The escaped string.
 */
/* #export */ function quoteString(str) {
return str.replace(/([\\\.\+\*\?\[\^\]\$\(\)\{\}\=\!\<\>\|\:])/g, '\\$1');
}
```

```
/**
 * Calls |callback| and stops listening the first time any event in |eventNames|
 * is triggered on |target|.
 * @param {!EventTarget} target
 * @param {!Array<string>|string} eventNames Array or space-delimited string of * event
names to listen to (e.g. 'click mousedown').
 * @param {function(!Event)} callback Called at most once. The
 * optional return value is passed on by the listener.
 */
/* #export */ function listenOnce(target, eventNames, callback) { if (!
Array.isArray(eventNames)) {
eventNames = eventNames.split(/ +/); }
const removeAllAndCallCallback = function(event)
{ eventNames.forEach(function(eventName) {
target.removeEventListener(eventName, removeAllAndCallCallback, false); });
return callback(event); };
eventNames.forEach(function(eventName) { target.addEventListener(eventName,
removeAllAndCallCallback, false);
}); }
/**
 * @param {!Event} e
 * @return {boolean} Whether a modifier key was down when processing |e|. */
/* #export */ function hasKeyModifiers(e) {
return !!(e.altKey || e.ctrlKey || e.metaKey || e.shiftKey);
}
/**
 * @param {!Element} el
 * @return {boolean} Whether the element is interactive via text input. */
/* #export */ function isTextInputElement(el) {
return el.tagName === 'INPUT' || el.tagName === 'TEXTAREA';
}
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————
VM164:2134 crbug/1173575, non-JS module files deprecated. (anonymous) @
VM164:2134
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
/**
```

- * @fileoverview This file defines a singleton which provides access to all data

- * that is available as soon as the page's resources are loaded (before DOM

- * content has finished loading). This data includes both localized strings

and

-        * any data that is important to have ready from a very early stage (e.g. things

-        * that must be displayed right away). *

-        * Note that loadTimeData is not guaranteed to be consistent between page

-        * refreshes (https://crbug.com/740629) and should not contain values that might

-        * change if the page is re-opened later. */
  /** @type {!LoadTimeData} */
  // eslint-disable-next-line no-var var loadTimeData;
  class LoadTimeData { constructor() {
  /** @type {?Object} */
  this.data_ = null; }
  /**
  * Sets the backing object.

*
* Note that there is no getter for |data_| to discourage abuse of the form: *
* var value = loadTimeData.data()['key'];
*
* @param {Object} value The de-serialized page data.
*/
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {
return id in this.data_;
}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value = this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;

```
}
/**
 * As above, but also makes sure that the value is a string.
 * @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
 */
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
 * Returns a formatted localized string where $1 to $9 are replaced by the * second to
the tenth argument.
 *      @param {string} id The ID of the string we want.

 *      @param {...(string|number)} var_args The extra values to include in
the

 *      formatted output.

 *      @return {string} The formatted string. */
getStringF(id, var_args) {
const value = this.getString(id); if (!value) {
return ''; }
const args = Array.prototype.slice.call(arguments); args[0] = value;
return this.substituteString.apply(this, args);
}
/**

 *      Returns a formatted localized string where $1 to $9 are replaced by the

 *      second to the tenth argument. Any standalone $ signs must be
escaped as

 *      $$.

* * * * * */
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted
output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
```

} /**
Returns a formatted string where $1 to $9 are replaced by the second to tenth argument, split apart into a list of pieces describing how the substitution was performed. Any standalone $ signs must be escaped as $$. @param {string} label A localized string to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted string pieces. getSubstitutedStringPieces(label, var_args) {
* * * * * * * * * */
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9 pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
[]).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
* As above, but also makes sure that the value is a boolean.
* @param {string} id The key that identifies the desired boolean. * @return {boolean} The corresponding boolean value.
*/
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type {boolean} */ (value);
}
/**
* As above, but also makes sure that the value is an integer.
* @param {string} id The key that identifies the desired number. * @return {number} The corresponding number value.
*/
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type {number} */ (value);
}
/**
* Override values in loadTimeData with the values found in |replacements|.

```
 * @param {Object} replacements The dictionary object of keys to replace.
 */ overrideValues(replacements) {
expect(
typeof replacements === 'object', 'Replacements must be a dictionary object.');
for (const key in replacements) { this.data_[key] = replacements[key];
} }
/**
 * Reset loadTimeData's data. Should only be used in tests.
 * @param {?Object} newData The data to restore to, when null restores to * unset state.
 */
resetForTesting(newData = null) { this.data_ = newData;
}
/**
 * @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
 * Checks condition, throws error message if expectation fails.
 * @param {*} condition The condition to check for truthiness.
 * @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
} }
/**
 * Checks that the given value has the given type.
 * @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
 * @param {string} type The type we expect |value| to be.
 */
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————————
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
/**
    •            *  @fileoverview This file defines a singleton which provides access to all
```

data

- * that is available as soon as the page's resources are loaded (before DOM

- * content has finished loading). This data includes both localized strings and

- * any data that is important to have ready from a very early stage (e.g. things

- * that must be displayed right away). *

- * Note that loadTimeData is not guaranteed to be consistent between page

- * refreshes (https://crbug.com/740629) and should not contain values that might

- * change if the page is re-opened later. */
  /** @type {!LoadTimeData} */
  // eslint-disable-next-line no-var var loadTimeData;
  class LoadTimeData { constructor() {
  /** @type {?Object} */
  this.data_ = null; }
  /**
  * Sets the backing object.

*
* Note that there is no getter for |data_| to discourage abuse of the form: *
* var value = loadTimeData.data()['key'];
*
* @param {Object} value The de-serialized page data.
*/
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {
return id in this.data_;
}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The

corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**
* As above, but also makes sure that the value is a string.
* @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
*/
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
* Returns a formatted localized string where $1 to $9 are replaced by the * second to
the tenth argument.

- * @param {string} id The ID of the string we want.

- * @param {...(string|number)} var_args The extra values to include in
   the

- * formatted output.

- * @return {string} The formatted string. */
   getStringF(id, var_args) {
   const value = this.getString(id); if (!value) {
   return ''; }
   const args = Array.prototype.slice.call(arguments); args[0] = value;
   return this.substituteString.apply(this, args);
   }
   /**

- * Returns a formatted localized string where $1 to $9 are replaced by the

- * second to the tenth argument. Any standalone $ signs must be
   escaped as

- * $$.

* * * * * */
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted
output.

```
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
} /**
Returns a formatted string where $1 to $9 are replaced by the second to tenth
argument, split apart into a list of pieces describing how the substitution was performed.
Any standalone $ signs must be escaped as $$. @param {string} label A localized string
to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the
formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted
string pieces. getSubstitutedStringPieces(label, var_args) {
* * * * * * * * * */
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9
pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
[]).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
* As above, but also makes sure that the value is a boolean.
* @param {string} id The key that identifies the desired boolean. * @return {boolean}
The corresponding boolean value.
*/
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type
{boolean} */ (value);
}
/**
* As above, but also makes sure that the value is an integer.
* @param {string} id The key that identifies the desired number. * @return {number} The
corresponding number value.
*/
getInteger(id) {
const value = this.getValue(id);
```

```
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
{number} */ (value);
}
/**
* Override values in loadTimeData with the values found in |replacements|.
* @param {Object} replacements The dictionary object of keys to replace.
*/ overrideValues(replacements) {
expect(
typeof replacements === 'object', 'Replacements must be a dictionary object.');
for (const key in replacements) { this.data_[key] = replacements[key];
}}
/**
* Reset loadTimeData's data. Should only be used in tests.
* @param {?Object} newData The data to restore to, when null restores to * unset state.
*/
resetForTesting(newData = null) { this.data_ = newData;
}
/**
* @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
}}
/**
* Checks condition, throws error message if expectation fails.
* @param {*} condition The condition to check for truthiness.
* @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
}}
/**
* Checks that the given value has the given type.
* @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
* @param {string} type The type we expect |value| to be.
*/
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
```

—————————————————————————
crbug/1173575, non-JS module files deprecated. (anonymous) @ VM31:929

/**
* @fileoverview Assertion support. */
/**
- * Verify |condition| is truthy and return |condition| if so.

- * @template T

- * @param {T} condition A condition to check for truthiness. Note that this

- * may be used to test whether a value is defined or not, and we don't want

- * to force a cast to Boolean.

- * @param {string=} opt_message A message to show on failure.

- * @return {T} A non-null |condition|.

- * @closurePrimitive {asserts.truthy}

- * @suppress {reportUnknownTypes} because T is not sufficiently
constrained. */
/* #export */ function assert(condition, opt_message) { if (!condition) {
let message = 'Assertion failed'; if (opt_message) {
message = message + ': ' + opt_message; }

const error = new Error(message); const global = function() {
const thisOrSelf = this || self;
/** @type {boolean} */ thisOrSelf.traceAssertionsForTesting; return thisOrSelf;
}();
if (global.traceAssertionsForTesting) {
console.warn(error.stack); }
throw error; }
return condition; }
/**
*
*
*
*
*
*

```
*
*
*
* *}
    ●           *  assertNotReached();

    ●           *  return document; *}

Call this from places in the code that should never be reached.
For example, handling all the values of enum with a switch() like this:
function getValueFromEnum(enum) { switch (enum) {
case ENUM_FIRST_OF_TWO: return first
case ENUM_LAST_OF_TWO: return last;
*
* This code should only be hit in the case of serious programmer error or
* * * * */
unexpected input.
@param {string=} opt_message A message to show when this is hit. @closurePrimitive
{asserts.fail}
/* #export */ function assertNotReached(opt_message) { assert(false, opt_message ||
'Unreachable code hit');
}
/**
* @param {*} value The value to check.
* @param {function(new: T, ...)} type A user-defined constructor.
* @param {string=} opt_message A message to show when this is hit. * @return {T}
* @template T
*/
/* #export */ function assertInstanceof(value, type, opt_message) {
// We don't use assert immediately here so that we avoid constructing an error //
message if we don't have to.
if (!(value instanceof type)) {
assertNotReached(
opt_message ||
'Value ' + value + ' is not a[n] ' + (type.name || typeof type));
}
return value; }
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
———————————————————————————
crbug/1173575, non-JS module files deprecated. (anonymous) @ VM32:1250
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// #import {assertInstanceof} from './assert.m.js'; // #import {dispatchSimpleEvent} from
'./cr.m.js';
/**
```

* Alias for document.getElementById. Found elements must be HTMLElements. *
@param {string} id The ID of the element to find.
* @return {HTMLElement} The found element or null if not found.
*/
/* #export */ function $(id) {
// Disable getElementById restriction here, since we are instructing other // places to re-use the $() that is defined here.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, HTMLElement) : null;
}
// TODO(devlin): This should return SVGElement, but closure compiler is missing // those externs.
/**
* Alias for document.getElementById. Found elements must be SVGElements.
* @param {string} id The ID of the element to find.
* @return {Element} The found element or null if not found. */
/* #export */ function getSVGElement(id) {
// Disable getElementById restriction here, since it is not suitable for SVG // elements.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, Element) : null;
}
/**
* @return {?Element} The currently focused element (including elements that are * behind a shadow root), or null if nothing is focused.
*/
/* #export */ function getDeepActiveElement() {
let a = document.activeElement;
while (a && a.shadowRoot && a.shadowRoot.activeElement) {
a = a.shadowRoot.activeElement; }
return a; }
//
/**
* @param {Node} el A node to search for ancestors with |className|.
* @param {string} className A class to search for.
* @return {Element} A node with class of |className| or null if none is found. */
/* #export */ function findAncestorByClass(el, className) { return /** @type {Element} */ (findAncestor(el, function(el) {
return el.classList && el.classList.contains(className); }));
}
/**
- * Return the first ancestor for which the {@code predicate} returns true.

- * @param {Node} node The node to check.

- * @param {function(Node):boolean} predicate The function that tests the

- * nodes.

- * @param {boolean=} includeShadowHosts

- * @return {Node} The found ancestor or null if not found. */
  /* #export */ function findAncestor(node, predicate, includeShadowHosts) { while (node !== null) {

```
if (predicate(node)) { break;
}
node = includeShadowHosts && node instanceof ShadowRoot ? node.host :
node.parentNode;
}
return node; }
/**
* Disables text selection and dragging, with optional callbacks to specify
* overrides.
* @param {function(Event):boolean=} opt_allowSelectStart Unless this function * is
defined and returns true, the onselectionstart event will be
* suppressed.
* @param {function(Event):boolean=} opt_allowDragStart Unless this function * is
defined and returns true, the ondragstart event will be suppressed.
*/
/* #export */ function disableTextSelectAndDrag( opt_allowSelectStart,
opt_allowDragStart) {
// Disable text selection. document.onselectstart = function(e) {
if (!(opt_allowSelectStart && opt_allowSelectStart.call(this, e))) { e.preventDefault();
} };
// Disable dragging. document.ondragstart = function(e) {
if (!(opt_allowDragStart && opt_allowDragStart.call(this, e))) { e.preventDefault();
} };
}
/**
* Check the directionality of the page.
* @return {boolean} True if Chrome is running an RTL UI. */
/* #export */ function isRTL() {
return document.documentElement.dir === 'rtl';
}
/**
* Get an element that's known to exist by its ID. We use this instead of just * calling
getElementById and not checking the result because this lets us
* satisfy the JSCompiler type system.
* @param {string} id The identifier name. * @return {!HTMLElement} the Element. */
/* #export */ function getRequiredElement(id) { return assertInstanceof(
```

$(id), HTMLElement, 'Missing required element: ' + id); }
/**

-       * Query an element that's known to exist by a selector. We use this instead of

-       * just calling querySelector and not checking the result because this lets us

-       * satisfy the JSCompiler type system.

-       * @param {string} selectors CSS selectors to query the element.

-       * @param {(!Document|!DocumentFragment|!Element)=} opt_context An optional

-       * context object for querySelector.

-       * @return {!HTMLElement} the Element. */
  /* #export */ function queryRequiredElement(selectors, opt_context) { const element = (opt_context || document).querySelector(selectors); return assertInstanceof(element, HTMLElement, 'Missing required element: ' + selectors); }
  /**

-       * Creates a new URL which is the old URL with a GET param of key=value.

-       * @param {string} url The base URL. There is not sanity checking on the URL so

-       * it must be passed in a proper format.

-       * @param {string} key The key of the param.

-       * @param {string} value The value of the param.

-       * @return {string} The new URL. */
  /* #export */ function appendParam(url, key, value) {
  const param = encodeURIComponent(key) + '=' + encodeURIComponent(value);
  if (url.indexOf('?') === -1) { return url + '?' + param;
  }
  return url + '&' + param; }
  /**
  * Creates an element of a specified type with a specified class name. * @param {string} type The node type.
  * @param {string} className The class name to use.
  * @return {Element} The created element.

```
*/
/* #export */ function createElementWithClassName(type, className) {
const elm = document.createElement(type); elm.className = className;
return elm;
}
/**
* transitionend does not always fire (e.g. when animation is aborted
* or when no paint happens during the animation). This function sets up
* a timer and emulate the event if it is not fired when the timer expires.
* @param {!HTMLElement} el The element to watch for transitionend.
* @param {number=} opt_timeOut The maximum wait time in milliseconds for the *
transitionend to happen. If not specified, it is fetched from |el|
* using the transitionDuration style value.
*/
/* #export */ function ensureTransitionEndEvent(el, opt_timeOut) { if (opt_timeOut ===
undefined) {
const style = getComputedStyle(el);
opt_timeOut = parseFloat(style.transitionDuration) * 1000;
// Give an additional 50ms buffer for the animation to complete.
opt_timeOut += 50; }
let fired = false; el.addEventListener('transitionend', function f(e) {
el.removeEventListener('transitionend', f);
fired = true; });
window.setTimeout(function() { if (!fired) {
cr.dispatchSimpleEvent(el, 'transitionend', true); }
}, opt_timeOut); }
/**
* Alias for document.scrollTop getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The Y document scroll offset.
*/
/* #export */ function scrollTopForDocument(doc) {
return doc.documentElement.scrollTop || doc.body.scrollTop;
}
/**
* Alias for document.scrollTop setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target Y scroll offset.
*/
/* #export */ function setScrollTopForDocument(doc, value)
{ doc.documentElement.scrollTop = doc.body.scrollTop = value;
}
/**
```

```
 * Alias for document.scrollLeft getter.
 * @param {!HTMLDocument} doc The document node where information will be *
queried from.
 * @return {number} The X document scroll offset.
 */
/* #export */ function scrollLeftForDocument(doc) {
return doc.documentElement.scrollLeft || doc.body.scrollLeft;
}
/**
 * Alias for document.scrollLeft setter.
 * @param {!HTMLDocument} doc The document node where information will be *
queried from.
 * @param {number} value The target X scroll offset.
 */
/* #export */ function setScrollLeftForDocument(doc, value)
{ doc.documentElement.scrollLeft = doc.body.scrollLeft = value;
}
/**
 * Replaces '&', '<', '>', '"', and ''' characters with their HTML encoding.
 * @param {string} original The original string.
 * @return {string} The string with all the characters mentioned above replaced. */
/* #export */ function HTMLEscape(original) { return original.replace(/&/g, '&amp;')
.replace(/</g, '&lt;') .replace(/>/g, '&gt;') .replace(/"/g, '&quot;') .replace(/'/g, '&#39;');
}
/**
 * Shortens the provided string (if necessary) to a string of length at most
 * |maxLength|.
 * @param {string} original The original string.
 * @param {number} maxLength The maximum length allowed for the string. * @return
{string} The original string if its length does not exceed
 * |maxLength|. Otherwise the first |maxLength| - 1 characters with '...'
 * appended.
 */
/* #export */ function elide(original, maxLength) { if (original.length <= maxLength) {
return original; }
return original.substring(0, maxLength - 1) + '\u2026'; }
/**
 * Quote a string so it can be used in a regular expression. * @param {string} str The
source string.
 * @return {string} The escaped string.
 */
/* #export */ function quoteString(str) {
return str.replace(/([\\\.\+\*\?\[\^\]\$\(\)\{\}\=\!\<\>\|\:])/g, '\\$1');
}
/**
 * Calls |callback| and stops listening the first time any event in |eventNames|
```

* is triggered on |target|.
* @param {!EventTarget} target
* @param {!Array<string>|string} eventNames Array or space-delimited string of * event names to listen to (e.g. 'click mousedown').
* @param {function(!Event)} callback Called at most once. The
* optional return value is passed on by the listener.
*/
/* #export */ function listenOnce(target, eventNames, callback) { if (!Array.isArray(eventNames)) {
eventNames = eventNames.split(/ +/); }
const removeAllAndCallCallback = function(event)
{ eventNames.forEach(function(eventName) {
target.removeEventListener(eventName, removeAllAndCallCallback, false); });
return callback(event); };
eventNames.forEach(function(eventName) { target.addEventListener(eventName, removeAllAndCallCallback, false);
}); }
/**
* @param {!Event} e
* @return {boolean} Whether a modifier key was down when processing |e|. */
/* #export */ function hasKeyModifiers(e) {
return !!(e.altKey || e.ctrlKey || e.metaKey || e.shiftKey);
}
/**
* @param {!Element} el
* @return {boolean} Whether the element is interactive via text input. */
/* #export */ function isTextInputElement(el) {
return el.tagName === 'INPUT' || el.tagName === 'TEXTAREA';
}
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————————
crbug/1173575, non-JS module files deprecated. (anonymous) @ VM40:2134
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the LICENSE file.
/**

- * @fileoverview This file defines a singleton which provides access to all data

- * that is available as soon as the page's resources are loaded (before DOM

- * content has finished loading). This data includes both localized strings and

- * any data that is important to have ready from a very early stage (e.g.

things

- * that must be displayed right away). *

- * Note that loadTimeData is not guaranteed to be consistent between page

- * refreshes (https://crbug.com/740629) and should not contain values that might

- * change if the page is re-opened later. */
/** @type {!LoadTimeData} */
// eslint-disable-next-line no-var var loadTimeData;

class LoadTimeData { constructor() {
/** @type {?Object} */
this.data_ = null; }
/**

- * Sets the backing object. *

- * Note that there is no getter for |data_| to discourage abuse of the form:
  *

- * var value = loadTimeData.data()['key']; *

- * @param {Object} value The de-serialized page data. */
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {
return id in this.data_;
}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The
corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**

```
     * As above, but also makes sure that the value is a string.
     * @param {string} id The key that identifies the desired string. * @return {string}
       The corresponding string value.
     */
    getString(id) {

const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
 ●              *  Returns a formatted localized string where $1 to $9 are replaced by the

 ●              *  second to the tenth argument.

 ●              *  @param {string} id The ID of the string we want.

 ●              *  @param {...(string|number)} var_args The extra values to include in
     the

 ●              *  formatted output.

 ●              *  @return {string} The formatted string. */
    getStringF(id, var_args) {
    const value = this.getString(id); if (!value) {
    return ''; }
    const args = Array.prototype.slice.call(arguments); args[0] = value;
    return this.substituteString.apply(this, args);
    }
    /**

 ●              *  Returns a formatted localized string where $1 to $9 are replaced by the

 ●              *  second to the tenth argument. Any standalone $ signs must be
     escaped as

 ●              *  $$.

* * * * * */
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted
output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
```

```
    return m === '$$' ? '$' : varArgs[m[1]]; });
  }
  /**
   * Returns a formatted string where $1 to $9 are replaced by the second to
   * tenth argument, split apart into a list of pieces describing how the
   * substitution was performed. Any standalone $ signs must be escaped as $$.
   * @param {string} label A localized string to substitute through.
   * This is not an resource ID.
   * @param {...(string|number)} var_args The extra values to include in the * formatted
   output.
   * @return {!Array<!{value: string, arg: (null|string)}>} The formatted
   * string pieces.
   */
  getSubstitutedStringPieces(label, var_args) {
    const varArgs = arguments;
    // Split the string by separately matching all occurrences of $1-9 and of // non $1-9
    pieces.
    const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
    []).map(function(p) {
    // Pieces that are not $1-9 should be returned after replacing $$ // with $.
    if (!p.match(/^\$[1-9]$/)) {
    expect(
    (p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
    return {value: p.replace(/\$\$/g, '$'), arg: null}; }
    // Otherwise, return the substitution value.
    return {value: varArgs[p[1]], arg: p}; });
    return pieces; }
  /**
   * As above, but also makes sure that the value is a boolean.
   * @param {string} id The key that identifies the desired boolean. * @return {boolean}
   The corresponding boolean value.
   */
  getBoolean(id) {
    const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type
    {boolean} */ (value);
  }
  /**
   * As above, but also makes sure that the value is an integer.
   * @param {string} id The key that identifies the desired number. * @return {number} The
   corresponding number value.
   */
  getInteger(id) {
    const value = this.getValue(id);
    expectIsType(id, value, 'number');
    expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
    {number} */ (value);
```

```
}
/**
 * Override values in loadTimeData with the values found in |replacements|. * @param
 {Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) { expect(
typeof replacements === 'object',
'Replacements must be a dictionary object.'); for (const key in replacements) {
this.data_[key] = replacements[key]; }
}
/**
 * Reset loadTimeData's data. Should only be used in tests.
 * @param {?Object} newData The data to restore to, when null restores to * unset state.
 */
resetForTesting(newData = null) { this.data_ = newData;
}
/**
 * @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
 * Checks condition, throws error message if expectation fails.
 * @param {*} condition The condition to check for truthiness.
 * @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
}
}
/**
 * Checks that the given value has the given type.
 * @param {string} id The id of the value (only used for error message). * @param {*}
 value The value to check the type on.
 * @param {string} type The type we expect |value| to be.
 */
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————————
VM31:929 crbug/1173575, non-JS module files deprecated. (anonymous) @ VM31:929
```

/**
* @fileoverview Assertion support. */
/**

- * Verify |condition| is truthy and return |condition| if so.

- * @template T

- * @param {T} condition A condition to check for truthiness. Note that this

- * may be used to test whether a value is defined or not, and we don't
   want

- * to force a cast to Boolean.

- * @param {string=} opt_message A message to show on failure.

- * @return {T} A non-null |condition|.

- * @closurePrimitive {asserts.truthy}

- * @suppress {reportUnknownTypes} because T is not sufficiently
   constrained.

*/
/* #export */ function assert(condition, opt_message) {
if (!condition) {
let message = 'Assertion failed'; if (opt_message) {
message = message + ': ' + opt_message; }
const error = new Error(message); const global = function() {
const thisOrSelf = this || self;
/** @type {boolean} */ thisOrSelf.traceAssertionsForTesting; return thisOrSelf;
}();
if (global.traceAssertionsForTesting) {
console.warn(error.stack); }
throw error; }
return condition; }
/**
*
*
*
*
*
*

*
*
*
* *}

- 	* assertNotReached();

- 	* return document; *}

Call this from places in the code that should never be reached.
For example, handling all the values of enum with a switch() like this:
function getValueFromEnum(enum) { switch (enum) {
case ENUM_FIRST_OF_TWO: return first
case ENUM_LAST_OF_TWO: return last;
*
* This code should only be hit in the case of serious programmer error or
* * * * */
unexpected input.
@param {string=} opt_message A message to show when this is hit. @closurePrimitive
{asserts.fail}
/* #export */ function assertNotReached(opt_message) { assert(false, opt_message ||
'Unreachable code hit');
}
/**
* @param {*} value The value to check.
* @param {function(new: T, ...)} type A user-defined constructor.
* @param {string=} opt_message A message to show when this is hit. * @return {T}
* @template T
*/
/* #export */ function assertInstanceof(value, type, opt_message) {
// We don't use assert immediately here so that we avoid constructing an error //
message if we don't have to.
if (!(value instanceof type)) {
assertNotReached(
opt_message ||
'Value ' + value + ' is not a[n] ' + (type.name || typeof type));
}
return value; }
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————
crbug/1173575, non-JS module files deprecated. (anonymous) @ VM32:1250
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// #import {assertInstanceof} from './assert.m.js'; // #import {dispatchSimpleEvent} from
'./cr.m.js';
/**

* Alias for document.getElementById. Found elements must be HTMLElements. *
@param {string} id The ID of the element to find.
* @return {HTMLElement} The found element or null if not found.
*/
/* #export */ function $(id) {
// Disable getElementById restriction here, since we are instructing other // places to re-use the $() that is defined here.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, HTMLElement) : null;
}
// TODO(devlin): This should return SVGElement, but closure compiler is missing // those externs.
/**
* Alias for document.getElementById. Found elements must be SVGElements. *
@param {string} id The ID of the element to find.
* @return {Element} The found element or null if not found.
*/
/* #export */ function getSVGElement(id) {
// Disable getElementById restriction here, since it is not suitable for SVG // elements.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, Element) : null;
}
/**
* @return {?Element} The currently focused element (including elements that are *
behind a shadow root), or null if nothing is focused.
*/
/* #export */ function getDeepActiveElement() {
let a = document.activeElement;
while (a && a.shadowRoot && a.shadowRoot.activeElement) {
a = a.shadowRoot.activeElement; }
return a; }
//
/**
* @param {Node} el A node to search for ancestors with |className|.
* @param {string} className A class to search for.
* @return {Element} A node with class of |className| or null if none is found. */
/* #export */ function findAncestorByClass(el, className) { return /** @type {Element}
*/ (findAncestor(el, function(el) {
return el.classList && el.classList.contains(className); }));
}
/**
- * Return the first ancestor for which the {@code predicate} returns true.

- * @param {Node} node The node to check.

- * @param {function(Node):boolean} predicate The function that tests the

- * nodes.

* @param {boolean=} includeShadowHosts
* @return {Node} The found ancestor or null if not found. */
/* #export */ function findAncestor(node, predicate, includeShadowHosts) { while (node !
== null) {
if (predicate(node)) { break;
}
node = includeShadowHosts && node instanceof ShadowRoot ? node.host :
}
return node; }
node.parentNode;
/**
* Disables text selection and dragging, with optional callbacks to specify
* overrides.
* @param {function(Event):boolean=} opt_allowSelectStart Unless this function * is
defined and returns true, the onselectionstart event will be
* suppressed.
* @param {function(Event):boolean=} opt_allowDragStart Unless this function * is
defined and returns true, the ondragstart event will be suppressed.
*/
/* #export */ function disableTextSelectAndDrag(opt_allowSelectStart,
opt_allowDragStart) {
// Disable text selection. document.onselectstart = function(e) {
if (!(opt_allowSelectStart && opt_allowSelectStart.call(this, e))) { e.preventDefault();
} };
// Disable dragging. document.ondragstart = function(e) {
if (!(opt_allowDragStart && opt_allowDragStart.call(this, e))) { e.preventDefault();
} };
}
/**
* Check the directionality of the page.
* @return {boolean} True if Chrome is running an RTL UI. */
/* #export */ function isRTL() {
return document.documentElement.dir === 'rtl';
}
/**
* Get an element that's known to exist by its ID. We use this instead of just * calling
getElementById and not checking the result because this lets us
* satisfy the JSCompiler type system.
* @param {string} id The identifier name.
* @return {!HTMLElement} the Element.
*/

/* #export */ function getRequiredElement(id) { return assertInstanceof(
$(id), HTMLElement, 'Missing required element: ' + id); }
/**
- * Query an element that's known to exist by a selector. We use this instead of

- * just calling querySelector and not checking the result because this lets us

- * satisfy the JSCompiler type system.

- * @param {string} selectors CSS selectors to query the element.

- * @param {(!Document|!DocumentFragment|!Element)=} opt_context An optional

- * context object for querySelector.

- * @return {!HTMLElement} the Element. */
  /* #export */ function queryRequiredElement(selectors, opt_context) { const element
  = (opt_context || document).querySelector(selectors); return assertInstanceof(
  element, HTMLElement, 'Missing required element: ' + selectors); }
  /**

- * Creates a new URL which is the old URL with a GET param of key=value.

- * @param {string} url The base URL. There is not sanity checking on the URL so

- * it must be passed in a proper format.

- * @param {string} key The key of the param.

- * @param {string} value The value of the param.

- * @return {string} The new URL. */
  /* #export */ function appendParam(url, key, value) {
  const param = encodeURIComponent(key) + '=' + encodeURIComponent(value);
  if (url.indexOf('?') === -1) { return url + '?' + param;
  }
  return url + '&' + param; }

/**
* Creates an element of a specified type with a specified class name. * @param {string}
type The node type.

```
 * @param {string} className The class name to use.
 * @return {Element} The created element.
 */
/* #export */ function createElementWithClassName(type, className) { const elm =
document.createElement(type);
elm.className = className;
return elm;
}
/**
 * transitionend does not always fire (e.g. when animation is aborted
 * or when no paint happens during the animation). This function sets up
 * a timer and emulate the event if it is not fired when the timer expires.
 * @param {!HTMLElement} el The element to watch for transitionend.
 * @param {number=} opt_timeOut The maximum wait time in milliseconds for the *
transitionend to happen. If not specified, it is fetched from |el|
 * using the transitionDuration style value.
 */
/* #export */ function ensureTransitionEndEvent(el, opt_timeOut) { if (opt_timeOut ===
undefined) {
const style = getComputedStyle(el);
opt_timeOut = parseFloat(style.transitionDuration) * 1000;
// Give an additional 50ms buffer for the animation to complete.
opt_timeOut += 50; }
let fired = false; el.addEventListener('transitionend', function f(e) {
el.removeEventListener('transitionend', f);
fired = true; });
window.setTimeout(function() { if (!fired) {
cr.dispatchSimpleEvent(el, 'transitionend', true); }
}, opt_timeOut); }
/**
 * Alias for document.scrollTop getter.
 * @param {!HTMLDocument} doc The document node where information will be *
queried from.
 * @return {number} The Y document scroll offset.
 */
/* #export */ function scrollTopForDocument(doc) {
return doc.documentElement.scrollTop || doc.body.scrollTop; }
/**
 * Alias for document.scrollTop setter.
 * @param {!HTMLDocument} doc The document node where information will be *
queried from.
 * @param {number} value The target Y scroll offset.
 */
/* #export */ function setScrollTopForDocument(doc, value)
{ doc.documentElement.scrollTop = doc.body.scrollTop = value;
}
```

```
/**
 * Alias for document.scrollLeft getter.
 * @param {!HTMLDocument} doc The document node where information will be *
 queried from.
 * @return {number} The X document scroll offset.
 */
/* #export */ function scrollLeftForDocument(doc) {
return doc.documentElement.scrollLeft || doc.body.scrollLeft;
}
/**
 * Alias for document.scrollLeft setter.
 * @param {!HTMLDocument} doc The document node where information will be *
 queried from.
 * @param {number} value The target X scroll offset.
 */
/* #export */ function setScrollLeftForDocument(doc, value)
{ doc.documentElement.scrollLeft = doc.body.scrollLeft = value;
}
/**
 * Replaces '&', '<', '>', '"', and ''' characters with their HTML encoding.
 * @param {string} original The original string.
 * @return {string} The string with all the characters mentioned above replaced. */
/* #export */ function HTMLEscape(original) { return original.replace(/&/g, '&amp;')
.replace(/</g, '&lt;') .replace(/>/g, '&gt;') .replace(/"/g, '&quot;')
.replace(/'/g, '&#39;'); }
/**
 * Shortens the provided string (if necessary) to a string of length at most
 * |maxLength|.
 * @param {string} original The original string.
 * @param {number} maxLength The maximum length allowed for the string. * @return
 {string} The original string if its length does not exceed
 * |maxLength|. Otherwise the first |maxLength| - 1 characters with '...'
 * appended.
 */
/* #export */ function elide(original, maxLength) { if (original.length <= maxLength) {
return original; }
return original.substring(0, maxLength - 1) + '\u2026'; }
/**
 * Quote a string so it can be used in a regular expression. * @param {string} str The
 source string.
 * @return {string} The escaped string.
 */
/* #export */ function quoteString(str) {
return str.replace(/([\\\.\+\*\?\[\^\]\$\(\)\{\}\=\!\<\>\|\:])/g, '\\$1');
}
/**
```

```
* Calls |callback| and stops listening the first time any event in |eventNames|
* is triggered on |target|.
* @param {!EventTarget} target
* @param {!Array<string>|string} eventNames Array or space-delimited string of * event
names to listen to (e.g. 'click mousedown').
* @param {function(!Event)} callback Called at most once. The
* optional return value is passed on by the listener.
*/
/* #export */ function listenOnce(target, eventNames, callback) { if (!
Array.isArray(eventNames)) {
eventNames = eventNames.split(/ +/); }
const removeAllAndCallCallback = function(event)
{ eventNames.forEach(function(eventName) {
target.removeEventListener(eventName, removeAllAndCallCallback, false); });
return callback(event); };
eventNames.forEach(function(eventName) { target.addEventListener(eventName,
removeAllAndCallCallback, false);
}); }
/**
* @param {!Event} e
* @return {boolean} Whether a modifier key was down when processing |e|. */
/* #export */ function hasKeyModifiers(e) {
return !!(e.altKey || e.ctrlKey || e.metaKey || e.shiftKey);
}
/**
* @param {!Element} el
* @return {boolean} Whether the element is interactive via text input. */
/* #export */ function isTextInputElement(el) {
return el.tagName === 'INPUT' || el.tagName === 'TEXTAREA';
}
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————————
VM40:2134 crbug/1173575, non-JS module files deprecated. (anonymous) @
VM40:2134
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
/**
```

- *   @fileoverview This file defines a singleton which provides access to all
    data

- *   that is available as soon as the page's resources are loaded (before
    DOM

- *   content has finished loading). This data includes both localized strings
    and

- * any data that is important to have ready from a very early stage (e.g. things

- * that must be displayed right away). *

- * Note that loadTimeData is not guaranteed to be consistent between page

- * refreshes (https://crbug.com/740629) and should not contain values that might

- * change if the page is re-opened later. */

```
/** @type {!LoadTimeData} */
// eslint-disable-next-line no-var var loadTimeData;
class LoadTimeData { constructor() {
/** @type {?Object} */
this.data_ = null; }
/**
```

- * Sets the backing object. *

- * Note that there is no getter for |data_| to discourage abuse of the form:
  *

- * var value = loadTimeData.data()['key']; *

- * @param {Object} value The de-serialized page data. */
```
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {
return id in this.data_;
}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The
corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
```

```
    } /**
```

* As above, but also makes sure that the value is a string.
* @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
*/
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
-        * Returns a formatted localized string where $1 to $9 are replaced by the

-        * second to the tenth argument.

-        * @param {string} id The ID of the string we want.

-        * @param {...(string|number)} var_args The extra values to include in
   the

-        * formatted output.

-        * @return {string} The formatted string. */
   getStringF(id, var_args) {
   const value = this.getString(id); if (!value) {
   return ''; }
   const args = Array.prototype.slice.call(arguments); args[0] = value;
   return this.substituteString.apply(this, args);
   }
   /**

-        * Returns a formatted localized string where $1 to $9 are replaced by the

-        * second to the tenth argument. Any standalone $ signs must be
   escaped as

-        * $$.

* * * * * */
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted
output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {

```
    expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
    return m === '$$' ? '$' : varArgs[m[1]]; });
}
/**
 * * * * * * * * * * */
Returns a formatted string where $1 to $9 are replaced by the second to tenth
argument, split apart into a list of pieces describing how the substitution was performed.
Any standalone $ signs must be escaped as $$. @param {string} label A localized string
to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the
formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted
string pieces.
getSubstitutedStringPieces(label, var_args) {
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9
pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
[]).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
 * As above, but also makes sure that the value is a boolean.
 * @param {string} id The key that identifies the desired boolean. * @return {boolean}
The corresponding boolean value.
 */
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type
{boolean} */ (value);
} /**
 * As above, but also makes sure that the value is an integer.
 * @param {string} id The key that identifies the desired number. * @return {number} The
corresponding number value.
 */
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
{number} */ (value);
```

```
}
/**
* Override values in loadTimeData with the values found in |replacements|. * @param
{Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) { expect(
typeof replacements === 'object',
'Replacements must be a dictionary object.'); for (const key in replacements) {
this.data_[key] = replacements[key]; }
}
/**
* Reset loadTimeData's data. Should only be used in tests.
* @param {?Object} newData The data to restore to, when null restores to * unset state.
*/
resetForTesting(newData = null) { this.data_ = newData;
}
/**
* @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
* Checks condition, throws error message if expectation fails.
* @param {*} condition The condition to check for truthiness.
* @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
} }
/**
* Checks that the given value has the given type.
* @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
* @param {string} type The type we expect |value| to be.
*/
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
——————————————————————————
// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
```

LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h. const SB_BOX_CHECKED = 'boxchecked';
const SB_DISPLAY_CHECK_BOX = 'displaycheckbox';
// This sets up the Extended Safe Browsing Reporting opt-in, either for
// reporting malware or invalid certificate chains. Does nothing if the
// interstitial type is not SAFEBROWSING or SSL or CAPTIVE_PORTAL. function setupExtendedReportingCheckbox() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' && interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!loadTimeData.getBoolean(SB_DISPLAY_CHECK_BOX)) { return;
}
if ($('privacy-link')) { $('privacy-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY);
return false;
});
$('privacy-link').addEventListener('mousedown', function() {
return false; });
}
$('opt-in-checkbox').checked = loadTimeData.getBoolean(SB_BOX_CHECKED); $('extended-reporting-opt-in').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-opt-in';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-opt-in'; }
$('extended-reporting-opt-in').classList.add(className);
$('body').classList.add('extended-reporting-has-checkbox');
if ($('whitepaper-link')) { $('whitepaper-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER); });
}
$('opt-in-checkbox').addEventListener('click', function() { sendCommand($('opt-in-checkbox').checked ?
}); }
SecurityInterstitialCommandId.CMD_DO_REPORT :
SecurityInterstitialCommandId.CMD_DONT_REPORT);
————————————————————————————
// Copyright 2020 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h.
const SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE =

```javascript
'displayEnhancedProtectionMessage';
// This sets up the enhanced protection message. function
setupEnhancedProtectionMessage() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' &&
interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!
loadTimeData.getBoolean(SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE)) {
return; }
if ($('enhanced-protection-link')) { if (mobileNav) {
// To make sure the touch area of the link is larger than the
// minimum touch area for accessibility, make the whole block tappable. $('enhanced-
protection-message').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
ON_SETTINGS);
return false; });
} else {
$('enhanced-protection-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
ON_SETTINGS);
return false; });
} }
$('enhanced-protection-message').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-enhanced-protection-message';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-enhanced-protection-message'; }
$('enhanced-protection-message').classList.add(className); }
———————————————————————————
// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
let mobileNav = false;
/**
* For small screen mobile the navigation buttons are moved * below the advanced text.
*/
function onResize() {
const helpOuterBox = document.querySelector('#details');
const mainContent = document.querySelector('#main-content');
const mediaQuery = '(min-width: 240px) and (max-width: 420px) and ' +
'(min-height: 401px), ' +
'(max-height: 560px) and (min-height: 240px) and ' + '(min-width: 421px)';
const detailsHidden = helpOuterBox.classList.contains(HIDDEN_CLASS); const
runnerContainer = document.querySelector('.runner-container');
```

```
// Check for change in nav status.
if (mobileNav !== window.matchMedia(mediaQuery).matches) {
mobileNav = !mobileNav;
// Handle showing the top content / details sections according to state. if (mobileNav) {
mainContent.classList.toggle(HIDDEN_CLASS, !detailsHidden);
helpOuterBox.classList.toggle(HIDDEN_CLASS, detailsHidden); if (runnerContainer) {
runnerContainer.classList.toggle(HIDDEN_CLASS, !detailsHidden); }
} else if (!detailsHidden) {
// Non mobile nav with visible details. mainContent.classList.remove(HIDDEN_CLASS);
helpOuterBox.classList.remove(HIDDEN_CLASS); if (runnerContainer) {
runnerContainer.classList.remove(HIDDEN_CLASS); }
} }
}
function setupMobileNav() { window.addEventListener('resize', onResize); onResize();
}
document.addEventListener('DOMContentLoaded', setupMobileNav);
——————————————————————————
// Copyright 2017 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// This is the shared code for security interstitials. It is used for both SSL // interstitials
and Safe Browsing interstitials.
/**
```

- * @typedef {{

- * dontProceed: function(),

- * proceed: function(),

- * showMoreSection: function(),

- * openHelpCenter: function(),

- * openDiagnostic: function(),

- * reload: function(),

- * openDateSettings: function(),

- * openLogin: function(),

- * doReport: function(),

- * dontReport: function(),

- * openReportingPrivacy: function(),

-            * openWhitepaper: function(),

-            * reportPhishingError: function(),

-            * }} */
  // eslint-disable-next-line no-var var certificateErrorPageController;
  // Should match security_interstitials::SecurityInterstitialCommand /** @enum {number} */

```
const SecurityInterstitialCommandId = { CMD_DONT_PROCEED: 0, CMD_PROCEED:
1,
// Ways for user to get more information CMD_SHOW_MORE_SECTION: 2,
CMD_OPEN_HELP_CENTER: 3, CMD_OPEN_DIAGNOSTIC: 4,
// Primary button actions CMD_RELOAD: 5, CMD_OPEN_DATE_SETTINGS: 6,
CMD_OPEN_LOGIN: 7,
// Safe Browsing Extended Reporting CMD_DO_REPORT: 8, CMD_DONT_REPORT: 9,
CMD_OPEN_REPORTING_PRIVACY: 10, CMD_OPEN_WHITEPAPER: 11,
// Report a phishing error. CMD_REPORT_PHISHING_ERROR: 12,
// Open enhanced protection settings.
CMD_OPEN_ENHANCED_PROTECTION_SETTINGS: 13,
};
const HIDDEN_CLASS = 'hidden';
/**
* A convenience method for sending commands to the parent page.
* @param {SecurityInterstitialCommandId} cmd The command to send. */
function sendCommand(cmd) {
if (window.certificateErrorPageController) {
switch (cmd) {
case SecurityInterstitialCommandId.CMD_DONT_PROCEED:
certificateErrorPageController.dontProceed();
break;
case SecurityInterstitialCommandId.CMD_PROCEED:
certificateErrorPageController.proceed();
break;
case SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION:
certificateErrorPageController.showMoreSection();
break;
case SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER:
certificateErrorPageController.openHelpCenter();
break;
case SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC:
certificateErrorPageController.openDiagnostic(); break;
case SecurityInterstitialCommandId.CMD_RELOAD:
certificateErrorPageController.reload();
break;
```

```
case SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS:
certificateErrorPageController.openDateSettings();
break;
case SecurityInterstitialCommandId.CMD_OPEN_LOGIN:
certificateErrorPageController.openLogin();
break;
case SecurityInterstitialCommandId.CMD_DO_REPORT:
certificateErrorPageController.doReport();
break;
case SecurityInterstitialCommandId.CMD_DONT_REPORT:
certificateErrorPageController.dontReport();
break;
case SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY:
certificateErrorPageController.openReportingPrivacy();
break;
case SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER:
certificateErrorPageController.openWhitepaper();
break;
case SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR:
certificateErrorPageController.reportPhishingError();
break;
case
SecurityInterstitialCommandId.CMD_OPEN_ENHANCED_PROTECTION_SETTINGS:
certificateErrorPageController.openEnhancedProtectionSettings();
break; }
return; }
// window.domAutomationController.send(cmd); //
//
}
/**
* Call this to stop clicks on <a href="#"> links from scrolling to the top of * the page (and
possibly showing a # in the link).
*/
function preventDefaultOnPoundLinkClicks() { document.addEventListener('click',
function(e) {
const anchor = findAncestor(/** @type {Node} */ (e.target), function(el) { return
el.tagName === 'A';
});
// Use getAttribute() to prevent URL normalization. if (anchor &&
anchor.getAttribute('href') === '#') {
e.preventDefault(); }
}); }
// //
——————————————————————————
// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
```

LICENSE file.
// This is the shared code for the new (Chrome 37) security interstitials. It is // used for both SSL interstitials and Safe Browsing interstitials.
let expandedDetails = false; let keyPressState = 0;
/**
* This allows errors to be skippped by typing a secret phrase into the page. * @param {string} e The key that was just pressed.
*/
function handleKeypress(e) {
// HTTPS errors are serious and should not be ignored. For testing purposes, // other approaches are both safer and have fewer side-effects.
// See https://goo.gl/ZcZixP for more details.
const BYPASS_SEQUENCE = window.atob('dGhpc2lzdW5zYWZl');
if (BYPASS_SEQUENCE.charCodeAt(keyPressState) === e.keyCode) {
keyPressState++;
if (keyPressState === BYPASS_SEQUENCE.length) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED);
keyPressState = 0; }
} else { keyPressState = 0;
} }
/**
* This appends a piece of debugging information to the end of the warning. * When complete, the caller must also make the debugging div

-        * (error-debugging-info) visible.

-        * @param {string} title The name of this debugging field.

-        * @param {string} value The value of the debugging field.

-        * @param {boolean=} fixedWidth If true, the value field is displayed fixed

-        * width. */
    function appendDebuggingField(title, value, fixedWidth) {
    // The values input here are not trusted. Never use innerHTML on these // values!
    const spanTitle = document.createElement('span');
    spanTitle.classList.add('debugging-title');
    spanTitle.innerText = title + ': ';
    const spanValue = document.createElement('span');
    spanValue.classList.add('debugging-content');
    if (fixedWidth) {
    spanValue.classList.add('debugging-content-fixed-width'); }
    spanValue.innerText = value;
    const pElem = document.createElement('p'); pElem.classList.add('debugging-content'); pElem.appendChild(spanTitle); pElem.appendChild(spanValue); $('error-debugging-info').appendChild(pElem);
    }

```
function toggleDebuggingInfo() {
const hiddenDebug = $('error-debugging-info').classList.toggle(HIDDEN_CLASS); $
('error-code').setAttribute('aria-expanded', !hiddenDebug);
}
function setupEvents() {
const overridable = loadTimeData.getBoolean('overridable'); const interstitialType =
loadTimeData.getString('type');
const ssl = interstitialType === 'SSL';
const captivePortal = interstitialType === 'CAPTIVE_PORTAL'; const badClock =
ssl && loadTimeData.getBoolean('bad_clock'); const lookalike = interstitialType ===
'LOOKALIKE';
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
const originPolicy = interstitialType === 'ORIGIN_POLICY';
const blockedInterception = interstitialType === 'BLOCKED_INTERCEPTION';
const legacyTls = interstitialType == 'LEGACY_TLS';
const insecureForm = interstitialType == 'INSECURE_FORM'; const httpsOnly =
interstitialType == 'HTTPS_ONLY';

const hidePrimaryButton = loadTimeData.getBoolean('hide_primary_button'); const
showRecurrentErrorParagraph = loadTimeData.getBoolean(
'show_recurrent_error_paragraph');
if (ssl || originPolicy || blockedInterception || legacyTls) { $
('body').classList.add(badClock ? 'bad-clock' : 'ssl');
if (loadTimeData.valueExists('errorCode')) {
$('error-code').textContent = loadTimeData.getString('errorCode');
$('error-code').classList.remove(HIDDEN_CLASS); }
```

- `} else if (captivePortal) { $('body').classList.add('captive-portal');`

- `} else if (billing) { $('body').classList.add('safe-browsing-billing');`

- `} else if (lookalike) { $('body').classList.add('lookalike-url');`

- `} else if (insecureForm) { $('body').classList.add('insecure-form');`

- `} else if (httpsOnly) { $('body').classList.add('https-only');`

- `} else {`
```
$('body').classList.add('safe-browsing');
// Override the default theme color. document.querySelector('meta[name=theme-
color]').setAttribute('content',
'rgb(217, 48, 37)'); }
$('icon').classList.add('icon');
if (legacyTls) { $('icon').classList.add('legacy-tls');
}
if (hidePrimaryButton) { $('primary-button').classList.add(HIDDEN_CLASS);
```

```
} else {
$('primary-button').addEventListener('click', function() {
switch (interstitialType) { case 'CAPTIVE_PORTAL':
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_LOGIN); break;
case 'SSL':
case 'LEGACY_TLS':
if (badClock) {

sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS); } else
if (overridable) {
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); } else {
sendCommand(SecurityInterstitialCommandId.CMD_RELOAD); }
break;
case 'SAFEBROWSING': case 'ORIGIN_POLICY':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
break;
case 'HTTPS_ONLY': case 'INSECURE_FORM': case 'LOOKALIKE':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); break;
default:
throw 'Invalid interstitial type';
} });
}
if (lookalike || insecureForm || httpsOnly) {
const proceedButton = 'proceed-button'; $
(proceedButton).classList.remove(HIDDEN_CLASS); $(proceedButton).textContent =
loadTimeData.getString('proceedButtonText'); $
(proceedButton).addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
}
if (lookalike) {
// Lookalike interstitials with a suggested URL have a link in the title:
// "Did you mean <link>example.com</link>?". Handle those clicks. Lookalike //
interstitails without a suggested URL don't have this link.
const dontProceedLink = 'dont-proceed-link';
if ($(dontProceedLink)) {
$(dontProceedLink).addEventListener('click', function(event)
{ sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
}); }
}
if (overridable) {
const overrideElement = billing ? 'proceed-button' : 'proceed-link';
// Captive portal page isn't overridable. $(overrideElement).addEventListener('click',
function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
if (ssl) { $(overrideElement).classList.add('small-link');
} else if (billing) { $(overrideElement).classList.remove(HIDDEN_CLASS); $
```

```
(overrideElement).textContent =
loadTimeData.getString('proceedButtonText'); }
} else if (!ssl) { $('final-paragraph').classList.add(HIDDEN_CLASS);
}
if (!ssl || !showRecurrentErrorParagraph) { $('recurrent-error-
message').classList.add(HIDDEN_CLASS);
} else { $('body').classList.add('showing-recurrent-error-message');
}
if ($('diagnostic-link')) { $('diagnostic-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC); });
}
if ($('learn-more-link')) { $('learn-more-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER); });
}
if (captivePortal || billing || lookalike || insecureForm || httpsOnly) { // Captive portal,
billing, lookalike pages, insecure form, and
// HTTPS only mode interstitials don't have details buttons. $('details-
button').classList.add('hidden');
} else { $('details-button')
.setAttribute(
'aria-expanded', !$('details').classList.contains(HIDDEN_CLASS));
$('details-button').addEventListener('click', function(event) {
const hiddenDetails = $('details').classList.toggle(HIDDEN_CLASS); $('details-
button').setAttribute('aria-expanded', !hiddenDetails);
if (mobileNav) {
// Details appear over the main content on small screens. $('main-
content').classList.toggle(HIDDEN_CLASS, !hiddenDetails);
} else { $('main-content').classList.remove(HIDDEN_CLASS);
}
$('details-button').innerText = hiddenDetails ? loadTimeData.getString('openDetails') :
loadTimeData.getString('closeDetails');
if (!expandedDetails) {
// Record a histogram entry only the first time that details is opened.
sendCommand(SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION);
expandedDetails = true;
} });
}
if ($('report-error-link')) { $('report-error-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR); })
;
}
if (lookalike) { console.log(
'Chrome has determined that ' + loadTimeData.getString('lookalikeRequestHostname')
+ ' could be fake or fraudulent.\n\n' +
'If you believe this is shown in error please visit ' + 'https://g.co/chrome/lookalike-
warnings');
```

```
}
preventDefaultOnPoundLinkClicks(); setupExtendedReportingCheckbox();
setupEnhancedProtectionMessage(); setupSSLDebuggingInfo();
document.addEventListener('keypress', handleKeypress);
}
document.addEventListener('DOMContentLoaded', setupEvents);
```
————————————————————————
```
<div id="debugging">
<div id="error-code" class="error-code" role="button" aria-
expanded="false">net::ERR_CERT_DATE_INVALID</div>
<div id="error-debugging-info" class="hidden"><p class="debugging-
content"><span class="debugging-title">Subject: </span><span class="debugging-
content">accounts.shopify.com</span></p><p class="debugging-content"><span
class="debugging-title">Issuer: </span><span class="debugging-content">R3</span></
p><p class="debugging-content"><span class="debugging-title">Expires on: </
span><span class="debugging-content">Nov 13, 2021</span></p><p
class="debugging-content"><span class="debugging-title">Current date: </span><span
class="debugging-content">Oct 5, 2021</span></p><p class="debugging-
content"><span class="debugging-title">PEM encoded chain: </span><span
class="debugging-content debugging-content-fixed-width">-----BEGIN
CERTIFICATE-----<br>MIIF2DCCBMCgAwIBAgISA3lP
+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA<br>MDIxCzAJBgNVBAYTAlV
TMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQYDVQQD<br>EwJSMzAeFw
0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNVBAMT<br>FGFj
Y291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB<br>CgK
CAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw<br>gkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw<br>5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH<br>w2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/
vtIs<br>ErLP9noTy1wdKzXJYbu4lT6nVOSs
+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX<br>MwPm/Opq3qk3BnI5uZgyV
+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/
BAQD<br>AgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBg
NVHRMBAf8EAjAA<br>MB0GA1UdDgQWBBR/TRz5xF8Ma/
wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX<br>t1hWy65QCUDmH6+dixTCxjBVB
ggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGFWh0<br>dHA6Ly9yMy5vLmxlbmN
yLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkubGVu<br>Y3Iub3JnLzCByQYDV
R0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91<br>ZC5jb22CFGFjY291bnRzLnNob3B
pZnkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w<br>aWZ5Y2xvdWQuY29tgiZpZGVudGl0
eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk<br>LmNvbYIiaWRlbnRpdHktbWVyY2h
hbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp<br>dHkuc2hvcGlmeWNsb3VkLmNv
bTBMBgNVHSAERTBDMAgGBmeBDAECATA3BgsrBgEE<br>AYLfEwEBATAoMCYGC
CsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0Lm9y<br>ZzCCAQMGCisGA
QQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj+KMDa5oK+<br>2MsxtT/
```

TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx
HBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14lTkrE
AHYAfT7y+I//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA
RzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfglhAOGXW8pNxh
+q
AgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQ
BcKanD
DWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWIfOU
LxGoEd/k
SA0t9X94AFhu+FhBQnrssUH/D
+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
dWfgQcEhx3HInYVT
+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h
Z8Y/kug
+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5
MVXdjQTeKuziIyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr
/+lfVfwGqqjAuskK
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/
bWLiWnX1owDQYJKoZIhvcNAQELBQAw
TzELMAkGA1UEBhMCVVMxKTAnBgN
VBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQD
EwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDAwMDAw
WhcNMjUwOTE1MTYw
MDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChMNTGV0J3Mg
RW5jcnl
wdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
R5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCU
JCWH+jdx
sxPnHKzhm+/
b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
NHz6a4uPVy
mZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
Z3Vms/
EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG
/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggElMIIBBDAOBgNVHQ8BAf8EBAMC
AYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
Af8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8
GA1UdIwQYMBaA
FHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCY
wJDAiBggrBgEFBQcw
AoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EI
DAeMBygGqAYhhZodHRw
Oi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCA
YGZ4EMAQIBMA0GCysGAQQB
gt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQ
CFyk5HPqP3hUSFvNVneLKYY611TR6W
PTNlclQtgaDqw+34IL9fzLdwALduO/
ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
ikfmZW4/RvTZ8M6UK
+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
CkcheAmCJ8MqyJu
5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh
+Blm
lJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4
avAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2
yJMC6alLbBfODALZvYH7n7do1AZls4I9d1P4jnkDrQox
B3UqQ9hVl3LEKQ73xF1O
yK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
hCExroL1+7mryIkXPeF
M5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
HlUjr8gRsI3qfJOQFy/
9rKIJR0Y/8Omwt/
8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
MldlTTKB3zhThV1+XWYp6rjd5JW1zbVW
EkLNxE7GJThEUG3szgBVGP7pSWTUTsqX
nLRbwHOoq7hHwg==
-----END

CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFYDCCBEigAwIBAgIQQAF3ITfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFAD
A/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMD
E4MTQwM1ow
TzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNIY
3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwg
gIiMA0GCSqGSIb3DQEB
AQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHI
e+3LffOJCMbjzmV6B493XC
ov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL
wYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJIII1PoOK5PCm
+D
LtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P
+1dy70sntK
4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5
bHbvO5BieebbpJovJsXQEOEO3tkQjhb
7t/eo98flAgeYjzYIIefiN5YNNnWe+w5y
sR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ
Xmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4
FQsDj43QLwWy
PntKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
SLeCO5imfW
CKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/
H2ql
PRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/
7xk9TND
TwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
SwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUF
BzAChi9odHRwOi8vYXBwcy5pZGVudHJ1
c3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgz
LnA3YzAfBgNVHSMEGDAWgBTEp7Gkeyxx
+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
ATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3Qte
DEubGV0c2VuY3J5cHQu
b3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9
jcmwuaWRlbnRydXN0LmNvbS9E
U1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OB
BYEFHm0WeZ7tuXkAXOACIjlGlj26Ztu
MA0GCSqGSIb3DQEBCwUAA4IBAQAKcw
Bslm7/DILQrt2M51oGrS+o44+/
yQoDFVDC
5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7
G75K/
QosMrAdSW
9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cl2qBDLKobkx/
J3vWraV0T9VuG
WCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6U
SbW1FzXSLr9O
he8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE
+FiYruq9RUWhiF1myv4Q6W
+CyBFC
Dfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5

END CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE
0MDExNVow
PzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ
28uMRcwFQYDVQQD
Ew5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNA
QEBBQADggEPADCCAQoCggEB
AN
+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/

IUmTrE4O
rz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
OLl5CjH9
UL2AZd
+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
xiqKqy69cK3FCx
olkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/
yUFw
7BZy1SbsOFU5Q9D8/
RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
aeQQmxkqtilX4+U9m
5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV
HQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG
SIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNI+DT69
ikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX
+5v3gTt23ADq1cEmv8uXr
AvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbu
mR9YbK
+rlmM6pZW87ipxZz
R8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5
JDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo
Ob8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ<b
r>-----END CERTIFICATE-----
</p><p class="debugging-content">Certificate Transparency: <span class="debugging-
content">

SCT Cloudflare 'Nimbus2021' Log (Embedded in certificate,
Verified)

SCT Google 'Xenon2021' log (Embedded in certificate, Verified)</
span></p></div>
</div> ————————————————————————
<div id="error-code" class="error-code" role="button" aria-
expanded="false">net::ERR_CERT_DATE_INVALID</div>
<div id="error-debugging-info" class="hidden"><p class="debugging-content">Subject: <span class="debugging-
content">accounts.shopify.com</p><p class="debugging-content">Issuer: R3</
p><p class="debugging-content">Expires on: </
span>Nov 13, 2021</p><p
class="debugging-content">Current date: Oct 5, 2021</p><p class="debugging-
content">PEM encoded chain: <span
class="debugging-content debugging-content-fixed-width">-----BEGIN
CERTIFICATE-----
MIIF2DCCBMCgAwIBAgISA3lP
+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA
MDIxCzAJBgNVBAYTAlV
TMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQYDVQQD
EwJSMzAeFw
0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNVBAMT
FGFj
Y291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgK
CAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw
gkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw
5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH
w2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/
vtIs
ErLP9noTy1wdKzXJYbu4lT6nVOSs

+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX
MwPm/Opq3qk3BnI5uZgyV
+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/
BAQD
AgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBg
NVHRMBAf8EAjAA
MB0GA1UdDgQWBBR/TRz5xF8Ma/
wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX
t1hWy65QCUDmH6+dixTCxjBVB
ggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGFWh0
dHA6Ly9yMy5vLmxlbmN
yLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkubGVu
Y3Iub3JnLzCByQYDV
R0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91
ZC5jb22CFGFjY291bnRzLnNob3B
pZnkuY29tgh9jdXN0b21IcnMtZGVtby5zaG9w
aWZ5Y2xvdWQuY29tgiZpZGVudGl0
eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk
LmNvbYIiaWRlbnRpdHktbWVyY2h
hbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp
dHkuc2hvcGlmeWNsb3VkLmNv
bTBMBgNVHSAERTBDMAgGBmeBDAECATA3BgsrBgEE
AYLfEwEBATAoMCYGC
CsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0Lm9y
ZzCCAQMGCisGA
QQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj+KMDa5oK+
2MsxtT/
TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx
HBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14lTkrE
AHYAfT7y+I//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA
RzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh
+q
AgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQ
BcKanD
DWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWIfOU
LxGoEd/k
SA0t9X94AFhu+FhBQnrssUH/D
+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
dWfgQcEhx3HInYVT
+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h
Z8Y/kug
+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5
MVXdjQTeKuziIyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3I7Z70dX1sEwUXQR8Wr
/+lfVfwGqqjAuskK
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/
bWLiWnX1owDQYJKoZIhvcNAQELBQAw
TzELMAkGA1UEBhMCVVMxKTAnBgN
VBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQD
EwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDAwMDAw
WhcNMjUwOTE1MTYw
MDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChMNTGV0J3Mg
RW5jcnl
wdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
R5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCU
JCWH+jdx
sxPnHKzhm+/
b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
NHz6a4uPVy
mZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
Z3Vms/
EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG
/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
AYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
Af8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8
GA1UdIwQYMBaA
FHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCY
wJDAiBggrBgEFBQcw
AoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EI

DAeMBygGqAYhhZodHRw
Oi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCA
YGZ4EMAQIBMA0GCysGAQQB
gt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQ
CFyk5HPqP3hUSFvNVneLKYY611TR6W
PTNlclQtgaDqw+34IL9fzLdwALduO/
ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
ikfmZW4/RvTZ8M6UK
+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
CkcheAmCJ8MqyJu
5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh
+BIm
IJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4
avAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2
yJMC6alLbBfODALZvYH7n7do1AZls4I9d1P4jnkDrQox
B3UqQ9hVl3LEKQ73xF1O
yK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
hCExroL1+7mryIkXPeF
M5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
HlUjr8gRsl3qfJOQFy/
9rKIJR0Y/8Omwt/
8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
MldlTTKB3zhThV1+XWYp6rjd5JW1zbVW
EkLNxE7GJThEUG3szgBVGP7pSWTUTsqX
nLRbwHOoq7hHwg==
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFYDCCBEigAwIBAgIQQAF3ITfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFAD
A/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTlxMDEyMDE5MTQwM1oXDTI0MDkzMD
E4MTQwM1ow
TzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY
3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwg
gIiMA0GCSqGSIb3DQEB
AQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHI
e+3LffOJCMbjzmV6B493XC
ov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL
wYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlll1PoOK5PCm
+D
LtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P
+1dy70sntK
4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5
bHbvO5BieebbpJovJsXQEOEO3tkQjhb
7t/eo98flAgeYjzYllefiN5YNNnWe+w5y
sR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ
Xmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4
FQsDj43QLwWy
PntKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
SLeCO5imfW
CKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/
H2ql
PRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/
7xk9TND
TwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
SwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUF
BzAChi9odHRwOi8vYXBwcy5pZGVudHJ1
c3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgz
LnA3YzAfBgNVHSMEGDAWgBTEp7Gkeyxx
+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
ATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3Qte
DEubGV0c2VuY3J5cHQu
b3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9
jcmwuaWRlbnRydXN0LmNvbS9E
U1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OB
BYEFHm0WeZ7tuXkAXOACIjlGlj26Ztu
MA0GCSqGSIb3DQEBCwUAA4IBAQAKcw
Bslm7/DlLQrt2M51oGrS+o44+/
yQoDFVDC
5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7

G75K/
QosMrAdSW
9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
WCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6U
SbW1FzXSLr9O
he8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE
+FiYruq9RUWhiF1myv4Q6W
+CyBFC
Dfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5

END CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE
0MDExNVow
PzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ
28uMRcwFQYDVQQD
Ew5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNA
QEBBQADggEPADCCAQoCggEB
AN
+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O
rz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
OLl5CjH9
UL2AZd
+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
xiqKqy69cK3FCx
olkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/
yUFw
7BZy1SbsOFU5Q9D8/
RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
aeQQmxkqtilX4+U9m
5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV
HQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG
SIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNI+DT69
ikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX
+5v3gTt23ADq1cEmv8uXr
AvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbu
mR9YbK
+rlmM6pZW87ipxZz
R8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5
JDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo
Ob8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ<b
r>-----END CERTIFICATE-----
</p><p class="debugging-content">Certificate Transparency: <span class="debugging-
content">

SCT Cloudflare 'Nimbus2021' Log (Embedded in certificate,
Verified)

SCT Google 'Xenon2021' log (Embedded in certificate, Verified)</
span></p></div>
———————————————————————————
<div id="recurrent-error-message" class="hidden">
Warnings may be common while websites update their security. This should
improve soon.
</div>
<div id="extended-reporting-opt-in" class="hidden">
<label>
<div class="checkboxes">
<input type="checkbox" id="opt-in-checkbox">

```html
<span class="checkbox"></span>
</div>
<span id="opt-in-label">Help improve security on the web for everyone by
sending <a href="#" id="whitepaper-link">URLs of some pages you visit, limited system
information, and some page content</a> to Google. <a id="privacy-link"
href="#">Privacy policy</a></span>
</label> </div>
<span id="opt-in-label">Help improve security on the web for everyone by sending <a
href="#" id="whitepaper-link">URLs of some pages you visit, limited system information,
and some page content</a> to Google. <a id="privacy-link" href="#">Privacy policy</
a></span>
<div id="enhanced-protection-message" class="hidden"> <label>
<div class="icon"></div>
<div>To get Chrome's highest level of security, <a href="#" id="enhanced- protection-
link">turn on enhanced protection</a></div>
</label> </div>
```

————————————————————————————

```css
/* Copyright 2014 The Chromium Authors. All rights reserved.
```
- `* Use of this source code is governed by a BSD-style license that can be`

- `* found in the LICENSE file. */`

```
/* This file is dynamically processed by a C++ data source handler to fill in
```
- `* some per-platform/locale styles that dramatically alter the page. This is`

- `* done to reduce flicker, as JS may not run before the page is rendered.`
  `*`

- `* There are two ways to include this stylesheet:`

- `* 1. via its chrome://resources/ URL in HTML, i.e.: *`

- `* <link rel="stylesheet" href="chrome://resources/css/text_defaults.css">`
  `*`

- `* 2. via the webui::AppendWebUICSSTextDefaults() method to directly`
  append it

- `* to an HTML string.`

```css
* Otherwise its placeholders won't be expanded. */
body {
font-family: system-ui, sans-serif; font-size: 75%;
}
button {
font-family: system-ui, sans-serif;
```

} —

/**

- 		* @fileoverview This file defines a singleton which provides access to all
data

- 		* that is available as soon as the page's resources are loaded (before
DOM

- 		* content has finished loading). This data includes both localized strings
and

- 		* any data that is important to have ready from a very early stage (e.g.
things

- 		* that must be displayed right away). *

- 		* Note that loadTimeData is not guaranteed to be consistent between
page

- 		* refreshes (https://crbug.com/740629) and should not contain values
that might

- 		* change if the page is re-opened later. */
/** @type {!LoadTimeData} */
// eslint-disable-next-line no-var var loadTimeData;
class LoadTimeData { constructor() {
/** @type {?Object} */
this.data_ = null; }
/**
* Sets the backing object.
*
* Note that there is no getter for |data_| to discourage abuse of the form: *

* var value = loadTimeData.data()['key'];
*
* @param {Object} value The de-serialized page data. */
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {

```
return id in this.data_;
}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The
corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**
* As above, but also makes sure that the value is a string.
* @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
*/
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
```

* Returns a formatted localized string where $1 to $9 are replaced by the

* second to the tenth argument.

* @param {string} id The ID of the string we want.

* @param {...(string|number)} var_args The extra values to include in the

* formatted output.

```
* @return {string} The formatted string.
*/
getStringF(id, var_args) {
const value = this.getString(id); if (!value) {
return ''; }
const args = Array.prototype.slice.call(arguments); args[0] = value;
return this.substituteString.apply(this, args);
}
/**
```

* Returns a formatted localized string where $1 to $9 are replaced by the

* second to the tenth argument. Any standalone $ signs must be escaped as

-         * $$.

```
* * * * * */
```
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
} /**
```
* * * * * * * * * */
```
Returns a formatted string where $1 to $9 are replaced by the second to tenth argument, split apart into a list of pieces describing how the substitution was performed. Any standalone $ signs must be escaped as $$. @param {string} label A localized string to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted string pieces.
getSubstitutedStringPieces(label, var_args) {
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9 pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) || []).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
* As above, but also makes sure that the value is a boolean.
* @param {string} id The key that identifies the desired boolean. * @return {boolean} The corresponding boolean value.
*/
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type {boolean} */ (value);
}

```
/**
 * As above, but also makes sure that the value is an integer.
 * @param {string} id The key that identifies the desired number. * @return {number} The
 corresponding number value.
 */
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
{number} */ (value);
}
/**
 * Override values in loadTimeData with the values found in |replacements|. * @param
{Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) {
expect(
typeof replacements === 'object', 'Replacements must be a dictionary object.');
for (const key in replacements) { this.data_[key] = replacements[key];
}}
/**
 * Reset loadTimeData's data. Should only be used in tests.
 * @param {?Object} newData The data to restore to, when null restores to * unset state.
 */
resetForTesting(newData = null) { this.data_ = newData;
}
/**
 * @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
}}
/**
 * Checks condition, throws error message if expectation fails.
 * @param {*} condition The condition to check for truthiness.
 * @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
}}
/**
 * Checks that the given value has the given type.
 * @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
 * @param {string} type The type we expect |value| to be.
 */
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type);
```

```
}
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
————————————————————————————
```

loadTimeData.data = {"bad_clock":false,"closeDetails":"Hide advanced","ct":"\n\nSCT
Cloudflare 'Nimbus2021' Log (Embedded in certificate, Verified)\n\nSCT Google
'Xenon2021' log (Embedded in certificate, Verified)","currentDate":"Oct 5,
2021","displayEnhancedProtectionMessage":false,"displaycheckbox":false,"enhancedPr
otectionMessage":"To get Chrome's highest level of security, \u003Ca href=\"#\" id=
\"enhanced-protection-link\">turn on enhanced protection\u003C/
a>","errorCode":"net::ERR_CERT_DATE_INVALID","errorType":1,"expirationDate":"Nov
13, 2021","explanationParagraph":"accounts.shopify.com normally uses encryption to
protect your information. When Chrome tried to connect to accounts.shopify.com this
time, the website sent back unusual and incorrect credentials. This may happen when
an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen
has interrupted the connection. Your information is still secure because Chrome stopped
the connection before any data was exchanged.","finalParagraph":"You cannot visit
accounts.shopify.com right now because the website uses HSTS. Network errors and
attacks are usually temporary, so this page will probably work
later.","fontfamily":"system-ui, sans-serif","fontsize":"75%","heading":"Your connection is
not
private","hide_primary_button":false,"issuer":"R3","language":"en","openDetails":"Advan
ced","optInLink":"Help improve security on the web for everyone by sending \u003Ca
href=\"#\" id=\"whitepaper-link\">URLs of some pages you visit, limited system
information, and some page content\u003C/a> to Google. \u003Ca id=\"privacy-link\"
href=\"#\">Privacy policy\u003C/a>","overridable":false,"pem":"-----BEGIN
CERTIFICATE-----\nMIIF2DCCBMCgAwIBAgISA3lP
+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA
\nMDIxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQY
DVQQD
\nEwJSMzAeFw0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNV
BAMT
\nFGFjY291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
\nCgKCAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw\ngkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw\n5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH\nw2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/vtIs
\nErLP9noTy1wdKzXJYbu4lT6nVOSs+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX
\nMwPm/Opq3qk3BnI5uZgyV+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/
BAQD

\nAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBgNVHRMB
Af8EAjAA\nMB0GA1UdDgQWBBR/TRz5xF8Ma/
wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX
\nt1hWy65QCUDmH6+dixTCxjBVBggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGF
Wh0\ndHA6Ly9yMy5vLmxlbmNyLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3lzLmkub
GVu\nY3Iub3JnLzCByQYDVR0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91\nZC5jb22CFGFjY291bnRzLnNob3BpZ
nkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w
\naWZ5Y2xvdWQuY29tgiZpZGVudGl0eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk
\nLmNvbYIiaWRlbnRpdHktbWVyY2hhbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp
\ndHkuc2hvcGlmeWNsb3VkLmNvbTBMBgNVHSAERTBDMAgGBmeBDAECATA3Bgsr
BgEE
\nAYLfEwEBATAoMCYGCCsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0L
m9y\nZzCCAQMGCisGAQQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj
+KMDa5oK+\n2MsxtT/TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx\nHBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14ITkrE\nAHYAfT7y+I//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA\nRzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh+q
\nAgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQBcKa
nD\nDWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWIfOULxGoEd/
k\nSA0t9X94AFhu+FhBQnrssUH/D+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
\ndWfgQcEhx3HInYVT+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h\nZ8Y/
kug+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5\nMVXdjQTeKuzilyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr\n/+lfVfwGqqjAuskK\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/bWLiWnX1owDQYJKoZlhvcNAQELBQAw
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDA
wMDAw
\nWhcNMjUwOTE1MTYwMDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChM
NTGV0J3Mg
\nRW5jcnlwdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEK\nAoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
\nR5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCUJCWH+jdx
\nsxPnHKzhm+/b5DtFUkWWQcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
\nNHz6a4uPVymZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
\nZ3Vms/EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG\n/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
\nAYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
\nAf8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrIAJQOYfr52LFMLGMB8GA1UdIwQY
MBaA
\nFHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCYwJDAiBggrBgEFBQc

w

\nAoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EIDAeMBygGqAYhhZodHRw

\nOi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCAYGZ4EMAQIBMA0GCysGAQQB

\ngt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQCFyk5HPqP3hUSFvNVneLKYY611T
R6W\nPTNlclQtgaDqw+34IL9fzLdwALduO/ZelN7kIJ+m74uyA+eitRY8kc607TkcC53wl
\nikfmZW4/RvTZ8M6UK+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
\nCkcheAmCJ8MqyJu5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh+Blm
\nlJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4\navAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2\nyJMC6alLbBfODALZvYH7n7do1AZls4I9d1P4jnkDrQoxB3
UqQ9hVl3LEKQ73xF1O\nyK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
\nhCExroL1+7mryIkXPeFM5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
\nHlUjr8gRsI3qfJOQFy/9rKIJR0Y/8Omwt/8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
\nMldlTTKB3zhThV1+XWYp6rjd5JW1zbVWEkLNxE7GJThEUG3szgBVGP7pSWTUTsq
X\nnLRbwHOoq7hHwg==\n-----END CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFYDCCBEigAwIBAgIQQAF3ITfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMDE4MTQw
M1ow
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwggIiMA0GCSqGSIb3D
QEB\nAQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHIe
+3LffOJCMbjzmV6B493XC\nov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL\nwYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlII1PoOK5PCm+D
\nLtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P+1dy70sntK
\n4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5\nbHbvO5BieebbpJovJsXQEOEO3tkQjhb7t/
eo98flAgeYjzYllefiN5YNNnWe+w5y\nsR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ\nXmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4\nFQsDj43QLwWyPn
tKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
\nSLeCO5imfWCKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/H2ql
\nPRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/7xk9TND
\nTwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
\nSwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUFBzAChi9odHRwOi8vYXBwcy5pZGVu
dHJ1\nc3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgzLnA3YzAfBgNVHSMEGDAWgBTEp7G
keyxx\n+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
\nATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3QteDEubGV0c2VuY3J5cHQ
u

\nb3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9jcmwuaWRlbnRydXN0LmNvbS9E
bS9E
\nU1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OBBYEFHm0WeZ7tuXkAXOACIjIGlj26
Ztu\nMA0GCSqGSIb3DQEBCwUAA4IBAQAKcwBslm7/DILQrt2M51oGrS+o44+/
yQoDFVDC
\n5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7G75K/
QosMrAdSW\n9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
\nWCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6USbW1FzXSLr9O
\nhe8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE+FiYruq9RUWhiF1myv4Q6W
+CyBFC\nDfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----\nMIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE0MDExN
Vow
\nPzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ28uMRcwFQYD
VQQD
\nEw5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQ
oCggEB\nAN+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O\nrz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1IcCJo6m67XMuegwGMoOifooUMM0RoOEq
\nOLl5CjH9UL2AZd+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
\nxiqKqy69cK3FCxolkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/yUFw
\n7BZy1SbsOFU5Q9D8/RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
\naeQQmxkqtilX4+U9m5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV\nHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG\nSIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNl+DT69\nikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX+5v3gTt23ADq1cEmv8uXr
\nAvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbumR9YbK+rlmM6pZW87ipxZz
\nR8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5\nNJDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo\nOb8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ\n-----
END CERTIFICATE-----\n","primaryButtonText":"Reload","primaryParagraph":"Attackers
might be trying to steal your information from \u003Cstrong>accounts.shopify.com
\u003C/strong> (for example, passwords, messages, or credit cards). \u003Ca href=\"#
\" id=\"learn-more-link\">Learn more\u003C/a>","recurrentErrorParagraph":"Warnings
may be common while websites update their security. This should improve
soon.","show_recurrent_error_paragraph":false,"subject":"accounts.shopify.com","tabTitl
e":"Privacy error","textdirection":"ltr","type":"SSL"};
————————————————————————————
<script>loadTimeData.data = {"bad_clock":false,"closeDetails":"Hide advanced","ct":"\n
\nSCT Cloudflare 'Nimbus2021' Log (Embedded in certificate, Verified)\n\nSCT Google
'Xenon2021' log (Embedded in certificate, Verified)","currentDate":"Oct 5,

2021","displayEnhancedProtectionMessage":false,"displaycheckbox":false,"enhancedPr otectionMessage":"To get Chrome's highest level of security, \u003Ca href=\"#\" id= \"enhanced-protection-link\">turn on enhanced protection\u003C/ a>","errorCode":"net::ERR_CERT_DATE_INVALID","errorType":1,"expirationDate":"Nov 13, 2021","explanationParagraph":"accounts.shopify.com normally uses encryption to protect your information. When Chrome tried to connect to accounts.shopify.com this time, the website sent back unusual and incorrect credentials. This may happen when an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.","finalParagraph":"You cannot visit accounts.shopify.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.","fontfamily":"system-ui, sans-serif","fontsize":"75%","heading":"Your connection is not private","hide_primary_button":false,"issuer":"R3","language":"en","openDetails":"Advan ced","optInLink":"Help improve security on the web for everyone by sending \u003Ca href=\"#\" id=\"whitepaper-link\">URLs of some pages you visit, limited system information, and some page content\u003C/a> to Google. \u003Ca id=\"privacy-link\" href=\"#\">Privacy policy\u003C/a>","overridable":false,"pem":"-----BEGIN CERTIFICATE-----\nMIIF2DCCBMCgAwIBAgISA3lP +wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA \nMDIxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQY DVQQD \nEwJSMzAeFw0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNV BAMT \nFGFjY291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB \nCgKCAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/ N7mdStXBNjXN9WAEbLSTGrw\ngkyShj1dbDXVJudJ7YdQXJrl/ KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw\n5Zgi0Sojj6d87SEqpS2i0Vz6V +hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH\nw2S2OFtEJxH3SSoKfsA/ ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/vtIs \nErLP9noTy1wdKzXJYbu4lT6nVOSs+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX \nMwPm/Opq3qk3BnI5uZgyV+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/ BAQD \nAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBgNVHRMB Af8EAjAA\nMB0GA1UdDgQWBBR/TRz5xF8Ma/ wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX \nt1hWy65QCUDmH6+dixTCxjBVBggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGGF Wh0\ndHA6Ly9yMy5vLmxlbmNyLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkub GVu\nY3Iub3JnLzCByQYDVR0RBIHBMIG +gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91\nZC5jb22CFGFjY291bnRzLnNob3BpZ nkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w \naWZ5Y2xvdWQuY29tgiZpZGVudGl0eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk \nLmNvbYIiaWRlbnRpdHktbWVyY2hhbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp \ndHkuc2hvcGlmeWNsb3VkLmNvbTMBgNVHSAERTBDMAgGBmeBDAECATA3Bgsr BgEE

\nAYLfEwEBATAoMCYGCCsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0L
m9y\nZzCCAQMGCisGAQQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj
+KMDa5oK+\n2MsxtT/TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx\nHBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14lTkrE\nAHYAfT7y+l//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA\nRzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh+q
\nAgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQBcKa
nD\nDWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWlfOULxGoEd/
k\nSA0t9X94AFhu+FhBQnrssUH/D+xQFKSbaZcQIB5j1nEFNdtpokxW5JyhjS5MXiYO
\ndWfgQcEhx3HInYVT+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h\nZ8Y/
kug+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5\nMVXdjQTeKuzilyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr\n/+lfVfwGqqjAuskK\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/bWLiWnX1owDQYJKoZIhvcNAQELBQAw
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDA
wMDAw
\nWhcNMjUwOTE1MTYwMDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChM
NTGV0J3Mg
\nRW5jcnlwdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEK\nAoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
\nR5QUVTVXjJ6oojkZ9Yl8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCUJCWH+jdx
\nsxPnHKzhm+/b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
\nNHz6a4uPVymZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
\nZ3Vms/EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG\n/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
\nAYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
\nAf8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8GA1UdIwQY
MBaA
\nFHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCYwJDAiBggrBgEFBQc
w
\nAoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EIDAeMBygGqAYhhZodH
Rw
\nOi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCAYGZ4EMAQIBMA0GCysGAQ
QB
\ngt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQCFyk5HPqP3hUSFvNVneLKYY611T
R6W\nPTNlclQtgaDqw+34IL9fzLdwALduO/ZelN7klJ+m74uyA+eitRY8kc607TkC53wl
\nikfmZW4/RvTZ8M6UK+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
\nCkcheAmCJ8MqyJu5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh+Blm
\nlJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4\nnavAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2\nnyJMC6alLbBfODALZvYH7n7do1AZls4l9d1P4jnkDrQoxB3

UqQ9hVl3LEKQ73xF1O\nyK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
\nhCExroL1+7mryIkXPeFM5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
\nHlUjr8gRsl3qfJOQFy/9rKIJR0Y/8Omwt/8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
\nMldlTTKB3zhThV1+XWYp6rjd5JW1zbVWEkLNxE7GJThEUG3szgBVGP7pSWTUTsq
X\nnLRbwHOoq7hHwg==\n-----END CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFYDCCBEigAwIBAgIQQAF3lTfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMDE4MTQw
M1ow
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwggIiMA0GCSqGSIb3D
QEB\nAQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHIe
+3LffOJCMbjzmV6B493XC\nov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL\nwYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlII1PoOK5PCm+D
\nLtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P+1dy70sntK
\n4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5\nbHbvO5BieebbpJovJsXQEOEO3tkQjhb7t/
eo98flAgeYjzYIlefiN5YNNnWe+w5y\nsR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ\nXmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4\nFQsDj43QLwWyPn
tKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
\nSLeCO5imfWCKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/H2ql
\nPRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/7xk9TND
\nTwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
\nSwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUFBzAChi9odHRwOi8vYXBwcy5pZGVu
dHJ1\nc3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgzLnA3YzAfBgNVHSMEGDAWgBTEp7G
keyxx\n+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
\nATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3QteDEubGV0c2VuY3J5cHQu
\nb3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9jcmwuaWRlbnRydXN0LmNv
bS9E
\nU1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OBBYEFHm0WeZ7tuXkAXOACIjIGlj26
Ztu\nMA0GCSqGSIb3DQEBCwUAA4IBAQAKcwBslm7/DlLQrt2M51oGrS+o44+/
yQoDFVDC
\n5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7G75K/
QosMrAdSW\n9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cl2qBDLKobkx/
J3vWraV0T9VuG
\nWCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6USbW1FzXSLr9O
\nhe8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE+FiYruq9RUWhiF1myv4Q6W
+CyBFC\nDfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----\nMIIDSjCCAjKgAwIBAgIQRK

+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE0MDExN
Vow
\nPzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ28uMRcwFQYD
VQQD
\nEw5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQ
oCggEB\nAN+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O\nrz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
\nOLl5CjH9UL2AZd+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dINb9
\nxiqKqy69cK3FCxolkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/yUFw
\n7BZy1SbsOFU5Q9D8/RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
\naeQQmxkqtilX4+U9m5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV\nHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG\nSIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNI+DT69\nikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX+5v3gTt23ADq1cEmv8uXr
\nAvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbumR9YbK+rlmM6pZW87ipxZz
\nR8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5\nJDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo\nOb8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ\n-----
END CERTIFICATE-----\n","primaryButtonText":"Reload","primaryParagraph":"Attackers
might be trying to steal your information from \u003Cstrong>accounts.shopify.com
\u003C/strong> (for example, passwords, messages, or credit cards). \u003Ca href=\"#
\" id=\"learn-more-link\">Learn more\u003C/a>","recurrentErrorParagraph":"Warnings
may be common while websites update their security. This should improve
soon.","show_recurrent_error_paragraph":false,"subject":"accounts.shopify.com","tabTitl
e":"Privacy error","textdirection":"ltr","type":"SSL"};</script>
—————————————————————————————
<html dir="ltr" lang="en"><head>
<meta charset="utf-8">
<meta name="color-scheme" content="light dark">
<meta name="theme-color" content="#fff">
<meta name="viewport" content="initial-scale=1, minimum-scale=1, width=device-
width">
<meta http-equiv="Content-Security-Policy" content="require-trusted-types-for 'script';
trusted-types;">
<title>Privacy error</title>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
a{
color: var(--link-color);
}

```css
body {
--background-color: #fff;
--error-code-color: var(--google-gray-700);
--google-blue-100: rgb(210, 227, 252);
--google-blue-300: rgb(138, 180, 248);
--google-blue-600: rgb(26, 115, 232);
--google-blue-700: rgb(25, 103, 210);
--google-gray-100: rgb(241, 243, 244);
--google-gray-300: rgb(218, 220, 224);
--google-gray-500: rgb(154, 160, 166);
--google-gray-50: rgb(248, 249, 250);
--google-gray-600: rgb(128, 134, 139);
--google-gray-700: rgb(95, 99, 104);
--google-gray-800: rgb(60, 64, 67);
--google-gray-900: rgb(32, 33, 36);
--heading-color: var(--google-gray-900);
--link-color: rgb(88, 88, 88); --popup-container-background-color: rgba(0,0,0,.65); --
primary-button-fill-color-active: var(--google-blue-700); --primary-button-fill-color: var(--
google-blue-600); --primary-button-text-color: #fff;
--quiet-background-color: rgb(247, 247, 247); --secondary-button-border-color: var(--
google-gray-500); --secondary-button-fill-color: #fff; --secondary-button-hover-border-
color: var(--google-gray-600); --secondary-button-hover-fill-color: var(--google-gray-50);
--secondary-button-text-color: var(--google-gray-700); --small-link-color: var(--google-
gray-700);
--text-color: var(--google-gray-700);
background: var(--background-color); color: var(--text-color);
word-wrap: break-word;
}
.nav-wrapper .secondary-button {
background: var(--secondary-button-fill-color);
border: 1px solid var(--secondary-button-border-color); color: var(--secondary-button-
text-color);
float: none;
margin: 0;
padding: 8px 16px;
}
.hidden { display: none;
}
html {
-webkit-text-size-adjust: 100%; font-size: 125%;
}
.icon {
background-repeat: no-repeat; background-size: 100%;
}
@media (prefers-color-scheme: dark) { body {
--background-color: var(--google-gray-900); --error-code-color: var(--google-gray-500);
```

--heading-color: var(--google-gray-500);
--link-color: var(--google-blue-300); --primary-button-fill-color-active: rgb(129, 162, 208);
--primary-button-fill-color: var(--google-blue-300); --primary-button-text-color: var(--google-gray-900); --quiet-background-color: var(--background-color); --secondary-button-border-color: var(--google-gray-700); --secondary-button-fill-color: var(--google-gray-900); --secondary-button-hover-fill-color: rgb(48, 51, 57); --secondary-button-text-color: var(--google-blue-300); --small-link-color: var(--google-blue-300);
--text-color: var(--google-gray-500); }
}
</style>

button {
border: 0;
border-radius: 4px;
box-sizing: border-box;
color: var(--primary-button-text-color);
cursor: pointer;
float: right;
font-size: .875em;
margin: 0;
padding: 8px 16px;
transition: box-shadow 150ms cubic-bezier(0.4, 0, 0.2, 1); user-select: none;
}
[dir='rtl'] button { float: left;
}
.bad-clock button, .captive-portal button, .https-only button, .insecure-form
button, .lookalike-url button, .main-frame-blocked button, .neterror button,
.pdf button,
.ssl button, .safe-browsing-billing button {
background: var(--primary-button-fill-color); }
button:active {
background: var(--primary-button-fill-color-active); outline: 0;
}
#debugging { display: inline; overflow: auto;
}
.debugging-content { line-height: 1em; margin-bottom: 0; margin-top: 1em;
}
.debugging-content-fixed-width { display: block;
font-family: monospace; font-size: 1.2em;
margin-top: 0.5em; }
.debugging-title { font-weight: bold;
}
#details {
margin: 0 0 50px;

```css
}
#details p:not(:first-of-type) { margin-top: 20px;
}
.secondary-button:active {
border-color: white;
box-shadow: 0 1px 2px 0 rgba(60, 64, 67, .3),
0 2px 6px 2px rgba(60, 64, 67, .15); }
.secondary-button:hover {
background: var(--secondary-button-hover-fill-color); border-color: var(--secondary-
button-hover-border-color); text-decoration: none;
}
.error-code {
color: var(--error-code-color); font-size: .8em;
margin-top: 12px; text-transform: uppercase;
}
#error-debugging-info { font-size: 0.8em;
}
h1 {
color: var(--heading-color); font-size: 1.6em; font-weight: normal; line-height: 1.25em;
margin-bottom: 16px;
}
h2 {
font-size: 1.2em; font-weight: normal;
}
.icon {
height: 72px; margin: 0 0 40px; width: 72px;
}
input[type=checkbox] { opacity: 0;
}
input[type=checkbox]:focus ~ .checkbox::after { outline: -webkit-focus-ring-color auto
5px;
}
.interstitial-wrapper { box-sizing: border-box; font-size: 1em; line-height: 1.6em; margin:
14vh auto 0; max-width: 600px; width: 100%;
}
#main-message > p { display: inline;
}
#extended-reporting-opt-in { font-size: .875em; margin-top: 32px;
}
#extended-reporting-opt-in label { display: grid; grid-template-columns: 1.8em 1fr;
position: relative;
}
#enhanced-protection-message { border-radius: 4px;
font-size: 1em;
margin-top: 32px;
padding: 10px 5px; }
```

```css
#enhanced-protection-message label { display: grid;
grid-template-columns: 2.5em 1fr; position: relative;
}
#enhanced-protection-message div { margin: 0.5em;
}
#enhanced-protection-message .icon { height: 1.5em;
vertical-align: middle;
width: 1.5em;
}
.nav-wrapper { margin-top: 51px;
}
.nav-wrapper::after { clear: both; content: '';
display: table; width: 100%;
}
.small-link {
color: var(--small-link-color); font-size: .875em;
}
.checkboxes { flex: 0 0 24px;
}
.checkbox {
--padding: .9em;
background: transparent;
display: block;
height: 1em;
left: -1em;
padding-inline-start: var(--padding); position: absolute;
right: 0;
top: -.5em;
width: 1em;
}
.checkbox::after { border: 1px solid white; border-radius: 2px; content: '';
height: 1em;
left: var(--padding); position: absolute;
top: var(--padding); width: 1em;
}
.checkbox::before { background: transparent; border: 2px solid white; border-inline-end-
width: 0; border-top-width: 0; content: '';
height: .2em;
left: calc(.3em + var(--padding)); opacity: 0;
position: absolute;
top: calc(.3em + var(--padding)); transform: rotate(-45deg);
width: .5em;
}
input[type=checkbox]:checked ~ .checkbox::before { opacity: 1;
}
#recurrent-error-message { background: #ededed; border-radius: 4px; margin-bottom:
```

```css
16px; margin-top: 12px; padding: 12px 16px;
}
.showing-recurrent-error-message #extended-reporting-opt-in { margin-top: 16px;
}
.showing-recurrent-error-message #enhanced-protection-message { margin-top: 16px;
}
@media (max-width: 700px) { .interstitial-wrapper {
padding: 0 10%; }
#error-debugging-info { overflow: auto;
} }
@media (max-width: 420px) { button,
[dir='rtl'] button,
.small-link {
float: none; font-size: .825em; font-weight: 500; margin: 0;
width: 100%;
}
button {
padding: 16px 24px;
}
#details {
margin: 20px 0 20px 0;
}
#details p:not(:first-of-type) { margin-top: 10px;
}
.secondary-button:not(.hidden) { display: block;
margin-top: 20px;
text-align: center;
width: 100%; }
.interstitial-wrapper { padding: 0 5%;
}
#extended-reporting-opt-in { margin-top: 24px;
}
#enhanced-protection-message { margin-top: 24px;
}
.nav-wrapper { margin-top: 30px;
} }
/**
* Mobile specific styling.
* Navigation buttons are anchored to the bottom of the screen.
* Details message replaces the top content in its own scrollable area. */
@media (max-width: 420px) { .nav-wrapper .secondary-button {
border: 0;
margin: 16px 0 0; margin-inline-end: 0; padding-bottom: 16px; padding-top: 16px;
} }
/* Fixed nav. */
@media (min-width: 240px) and (max-width: 420px) and
```

```css
(min-height: 401px),
(min-width: 421px) and (min-height: 240px) and (max-height: 560px) {
body .nav-wrapper {
background: var(--background-color);
bottom: 0;
box-shadow: 0 -12px 24px var(--background-color); left: 0;
margin: 0 auto;
max-width: 736px;
padding-inline-end: 24px;
padding-inline-start: 24px;
position: fixed;
right: 0;
width: 100%;
z-index: 2;
}
.interstitial-wrapper { max-width: 736px;
}
#details, #main-content {
padding-bottom: 40px; }
#details { padding-top: 5.5vh;
}
button.small-link {
color: var(--google-blue-600);
} }
@media (max-width: 420px) and (orientation: portrait), (max-height: 560px) {
body {
margin: 0 auto;
}
button,
[dir='rtl'] button,
button.small-link,
.nav-wrapper .secondary-button {
font-family: Roboto-Regular,Helvetica; font-size: .933em;
margin: 6px 0;
transform: translatez(0);
}
.nav-wrapper { box-sizing: border-box; padding-bottom: 8px; width: 100%;
}
#details {
box-sizing: border-box;
height: auto;
margin: 0;
opacity: 1;
transition: opacity 250ms cubic-bezier(0.4, 0, 0.2, 1);
}
#details.hidden, #main-content.hidden {
```

```css
height: 0;
opacity: 0; overflow: hidden; padding-bottom: 0; transition: none;
}
h1 {
font-size: 1.5em; margin-bottom: 8px;
}
.icon {
margin-bottom: 5.69vh;
}
.interstitial-wrapper { box-sizing: border-box; margin: 7vh auto 12px; padding: 0 24px;
position: relative;
}
.interstitial-wrapper p {
font-size: .95em; line-height: 1.61em; margin-top: 8px;
}
#main-content {
margin: 0;
transition: opacity 100ms cubic-bezier(0.4, 0, 0.2, 1);
}
.small-link { border: 0;
}
.suggested-left > #control-buttons, .suggested-right > #control-buttons {
float: none;
margin: 0; }
}
@media (min-width: 421px) and (min-height: 500px) and (max-height: 560px)
{ .interstitial-wrapper {
margin-top: 10vh; }
}
@media (min-height: 400px) and (orientation:portrait) { .interstitial-wrapper {
margin-bottom: 145px; }
}
@media (min-height: 299px) { .nav-wrapper {
padding-bottom: 16px; }
}
@media (max-height: 560px) and (min-height: 240px) and (orientation:landscape)
{ .extended-reporting-has-checkbox #details {
padding-bottom: 80px; }
}
@media (min-height: 500px) and (max-height: 650px) and (max-width: 414px) and
(orientation: portrait) { .interstitial-wrapper {
margin-top: 7vh; }
}
@media (min-height: 650px) and (max-width: 414px) and (orientation: portrait)
{ .interstitial-wrapper {
margin-top: 10vh; }
```

```
}
/* Small mobile screens. No fixed nav. */
@media (max-height: 400px) and (orientation: portrait),
(max-height: 239px) and (orientation: landscape),
(max-width: 419px) and (max-height: 399px) { .interstitial-wrapper {
display: flex; flex-direction: column; margin-bottom: 0;
}
#details {
flex: 1 1 auto; order: 0;
}
#main-content { flex: 1 1 auto; order: 0;
}
.nav-wrapper {
flex: 0 1 auto; margin-top: 8px; order: 1; padding-inline-end: 0; padding-inline-start: 0;
position: relative; width: 100%;
}
button,
.nav-wrapper .secondary-button {
padding: 16px 24px; }
button.small-link {
color: var(--google-blue-600);
} }
@media (max-width: 239px) and (orientation: portrait) { .nav-wrapper {
padding-inline-end: 0;
padding-inline-start: 0; }
} </style>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.bad-clock .icon {
background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAYAAABV7bNHAAAFo0lE
QVR4Xu3cS1OTVxwG8Ha6dsZNt/0S7ozX
+wUSGKN7yclvkJ2OiNcdbvwMfABXLS1VvLXFSMWUgFAh1oJICCEGq8UFp8+fPu87J
4S3vrmcvlfOceaZMKOSnN/8z/
+c95Yv3B8XFxcXFzNRZ89+rZLJJNKLDCAZdfp0Hini5zWJ/lzk5e+QAf7bpPzf/
yvKLkD0Il1lArCOqwazL70D65Xdub5RUagcA0kDJCYCJ8Hen5b22D8y5czvVmTN9gCk
RwHzwXvKe8t72wij1JWB6AFMIBDAPVZDPIJ/
Fth7zDWAeESD64LPIZ7ICBwAxZBFRlmURiUWLk0ymuCwrG8NtQyqKKfUVYG4RwP7
gs8pnbhsOAO4gapvlTluQzFaO+Uoy3nNMAayurlalcOqUKaSUudUKTa9dQN/
t3m0EiWOItX6fw6W8nUCDyFJHh5ktAMbUuh0yN4HtBPpWgGIxNYQUTSBhTC3Zccv
W3QNoawUR6Afk7p49ZpAwtuYPPHlsFUkFEehHAA0jy61GwthkjM1UTx8BIq2gIeDcQ
+7v3atKnZ2trqK+xs/n8JRFVBUEJL+CBGgYQD/
t369WWWomEMTZ0PgkAaQ8g6ik2RJyH+/
apxwD6BSm3tpLS9QPxTGDUFTQkTZrT6wGApIJGkKcHD6p38XjLzkzWfw6ZADb0oHv
```

AEaBHAPpZKohAv7YSCWOup3r6CRBpBX3v9R
+pHuSxAB04oDKI4EjGDh1qDRLGXA9Q1iagYQFigxagJwR6BpwxptIsEsYc
+roVANajBhpkcxag+3qDBs5TRMd5jowfPtws0rqMPUz1JAkQbQ8i0F1vBWP/
GQHOqEwvgWGywPmNWW0GCWMPA9RrA9Cg339YPQR6AhwBGtuEM47kjhxRE0jD
SBh7GKABC4CqGzSBZHplNCDA6PGBXhw9qt4nEo0ADYQBytgCJLvnB97yLs2ZQM
+Q59WV4+NM8nWqESSMPcwOOm8BEA8v2KC5QcxIgxYgTq0sgSY8HFaP4Ewjvx87
Vi9SPgxQ0QYgHqD6q9cIK2jU7z2sHIY4NUAzyF/hkYphptiaDUBjFy6oYa//
aLtnArHn1FaOjvMSkdfZsEgY+7YBqpTLKnvxonrI/jPiTS/
ijBPHAxKYFxrONDLD5I8f33j9AKQQQHZOsaVr1wSlBmkcSN7hxagGlPtM5Xg4swR6xX
wGqWhnk2aWr1/fEmni0iUfiNPLX84nQ+L8wbw+ceK/
kPKGlnnzSFNA8nbPOQBNalOKOEw1DiMwfuaQj11dDS7z3ChGmVIA0nRvr8oCaKPvb
MJ5ycxqOLObKudP4syfPCmpRcLYDRxqmMlKANLM5cvSf/
zqmUI4rQik9ZwAnDnkDbKA/
K0jYeyhD1ZtRnoFJKxSWuVwWjHEERjiyCsrR8N5iyu3i4iPhLEbON1hNuUApNdXrgAoo
HJqgHQchDiSApE
+dXXxdIeBE2am8+7GjS2R5oBU1ZCDp5WfBR2HweVtueaWNXDKNXqkeSARRlutiMP
404pZJJCHU0RwKanf8El786kEIL0BEisnsOfoOJICIzjLSLmjY5fFl33CZ/
XmzSCkwMqZD5hWgkOgnIELh/
Yhvb16Vc0RqXa1Ympx1EoikTZw6TnavA9AWujpERgdh0B+5bAp
+72npBKJHQZuXrAPqXD7dlDlEIdVw
+BGCKmePgO3v9iHBJzaymF0nCUNB9f3C6q7e6eBG6jsyofz57V9DrOpIRc0GMkKUun
s7DFwC56dwYGnvpTX9hzGwynF47wFz8BNnLYGx1ScUn64WlXjlOPxxY8Yk6HbgO1H
0nsOo+OsVRKJmNEbyW3PGpAAhGg9hwFOyj2KQCR/
WjG4TeaWe5hlE5LAlBG8bvUwi3sc6lN390blGMBxD9S5RzINxD3U6x4Ld18s4L6awt64
LzdxX4/zDxj9/IEueAvhAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAJAAAACQCAMAAADQmBKKAAACvl
BMVEUAAAD/gID/gID/VVX/VVX/Tk7/YmL/YGD/VVXzUVH/XV32UlL/W1v2T0//
WFj3UlL/UlL3UFD/WFjwTk7/U1P/U1PxTU3/V1fyT0//VFTzTk7/UlLwTU3/VVX0UFD/
VFT/VFT1Tk7/VVX/VFT/U1PyT0//VFT/U1PxTEz/UlLuS0v/U1P/UlL/VFT0T0//U1P0Tk7/
VFT/U1PuTU3/UlLzTU3/U1P/U1PwTEz/UlL/U1PvTU3/U1PxTU3/UlLzTk70Tk7/
U1PyTk7/U1PzTk7/U1P/U1P6UFD/UlLzTk7/U1P/U1PyTk7/U1PtTEz/UlLyTU3/U1P/UlL/
UlLxTk7/UlLvTEz/U1PvTU3/U1P/U1P/UlLxTEzxTU3zTU3/UlK7Ozu8Ozu8PDy9PDy
+PDy+PT2/PDy/PT3APDzAPT3BPj7BPj7CPT3CPj7DPT3DPj7EPj7EPz/FPj7FPz/
GPj7GPz/HPz/HQEDIPz/IQEDJPz/
JQEDKQEDKQUHLQEDLQUHMQEDMQUHNQUHNQkLOQUHOQkLOZWXPQUHPQk
LPZWXQQkLRQkLRQ0PSQkLSQ0PSZmbTQ0PTZmbUQ0PURETVQ0PVRETVaGjWR
ETWRUXXRETXRUXXaGjYRUXZRUXZaGjaRUXaRkbaaWnbRUXbRkbbaWncRkbdRk
bdaWneRkbeR0ffRkbfR0ffa2vgR0fga2vhR0fhSEjha2viR0fiSEjia2vjSEjjbGzkSEjkSUnkb
GzlSEjlSUnlbGzmSUnmbGznSUnnSkroSkrobW3pSkrqSkrqS0vqi4vrS0vriYnri4vsS0vsi
YntS0vtTEzuTEzvTEzwTEzwTU3w6OjxTU3x6OjyTU3y6Ojy6eny8vLz8/
P0Tk71Tk72Tk72cnL3T0/3cnL4T0/4cnL5T0/5c3P6T0/7UFD8UFD9UFD/
UlJJWZWgAAAAYXRSTlMAAgQGDA0NEBUWFhwcHR0fHyAgNDQ3ODg9PT4+QkJD
Q0ILS15fdHR1fHyEhlWGiIiJiYuVlaioqaurK+vuLm5u7u7wsLExMXGxszM0tTU2dna2t/
p7Ozt7fPz+fv+/v7+jD+tjQAACYhJREFUeAHs1cFqwjAcx/
G1FR0iIqKIFFFEUHKJQKIlRFKGUilSKVCmiHrKpCDuPHcbA99xtjA1+b7HLjmMkaeIu

+TzBl18C/xtFURRFURSFQ6bc6g0ns8Uq3u3i1WI2GfZa5cz/tKQafW
+NX629fiN11Rij1p3v8af9vFszrpRTGWxAZTOoyK8pdpZgsOwUpeaUxgSMyLgkLac6BZ
dpVUpO3QM3ry48JzdCIqOc0By9vUVC27YurscMIUBoiprHIhCCWEJGyvsQxhfwk5oxBI
qbCXM0m0AoYmuJrqgD4RyDvyftQgl3zduTDSBFcMvXU4ggSVTg2ieCNFGW4/8EkCh
g/keGC6lcg61Hc8Dg7cc76DkaU5ANnqAnliKb6V4QrqB7liLCcEXyMfiCHp4/QC3O0/
boPniDDi8MRb5OGWSBN+hwOL4yFFl0PSYB/
0LH0+UTtIhJ9WAhEgSdTufLF2iFNI92ByRa6PzIUPTNih21uFG1cQC/
Kewn6GWh0BdeSu1NKRRKS2+EUuiF2CKl7UWR3WabNHFmk84k2WSTndmsGyfZMc
42ziYmJhozbo0hxSVsMYRYFhEEFWt7Uau1V3q+hf+Ts2ZXMsmMnHk+wY//ec55zjn/d/
Heuc4DmqUg/4pr0XXnO+1Zwp2QP5B1LTrr
+D4lHKC3aQ8BFHxn3bXI6U37OuFPKBAUI4U/3b6yHf4TCIqrh3w
+vwCQVHQrmv4TcYFwJ3THHxIkSYqZLkUXpv7/
EC4QTmrf3J1QSAwDtFh2KZr2f3SOGzQXCNwNCbIkx5LpqjvRuSn/dbcIbw/
5fYGQIN6T5EQ6rX7qSnRr8q/fcYLi2vaz876AIIr34olESlEzzb+Iizo+EfQmJwie
+WAQLXQvGk
+mlEwma7kRvTHxv5dwJ3R7PiAIoixHk8mUmnErmvR7fJoXNHsbp3RQkMLRaHw5BY6
maV+4EJ2e8PK5xg2apaCIiJZOJJfVbDarZXNtZ9E1+zfREclNoi0UXJBjMbpiq/
BouZzWcRYdsQWd4AfNYY4tLOAQiidVdVWjnpyuf
+UoOmELusQP8gfQ0wuynEBCqwDldHh0w1F0yc5z6CY3aM6POSZIsryYSCtoIayXXsg
bht51EN08ZAM6SrhBPl9IFDDpFxfT6QxaGvkUUEbR6DuIjtqATvGD/
L67oihF5MTSUlrNaDoKHN0oFs2vydQ6ZXs141+yu4IQFumeX1LX1vKaRj2GUQSo9M1
/vqZd9SAhP0BhKYEOUjJrWr6gF3Tq2TBNs/SYTKmr454Zwg+ax6AXo/
F4Mqkqa3l4DJYPOGalskum1MwY6LAHIAQkShIGq6KghfI6Wy8EVCqVKpXyt2RyHR4
D/c8DUCAgRiJR3Dxw9UBLG6yfTXjMUqVarX1HJtaxMdBJD0BBQYhEpEQqqWBs5Kjl
+CeeKkDTRCfHQGc8ANGAZDmeSuGYph7WP0UmqtUatcb3ZEKd4X9v2CUUBAhjQ1l
ZwdgoIB54hhzkU6k1GvXmD67fHpc8AAnhSCQmJxQEIF1f3yiAM/
SUwUFA9Uaz2fzR7TS77AVIACiWximEMVYoDNeL5VODh3KshmUvujwGuuIBKBSOyA
Ap6iruHRs4DunuMoftzECWtbVl/URs6soY6C0PQO+KmPRoIUx6BASQud8/
aCDLagLUav1s9wM6BrrhAejJe1EJPa2q2bymb2zQfEbx1Fk
+8LTaNqIbXoJePt8XxZMJVVWyGKvY88P1QjtXWf8AZG21HrTbrV
+cQTxL9ur5vuj9ZAoLltHzBlasshcQPA14miyfdrvTeeawZJxN/
fuTA6LllJrVADKKFRQ81RryoR29hQKo1elsbz9zbOrLxCPRB
+oqNplxv1iulPfyoZrhBoMG+WwD1P3VadtfJF6JPqRzo3Afoio7EJt1ixaLp418HnZR/
xZdHAOdJ96JNGx6c7NC46Ee1Kh/Oggl+XS3d3ZekAN13nm4cog
+Wsce29xkGx7FNrw1BIHzcHunu7PTe/Ri6nA9SbwTPf14vWCW2X5nG77J8kE/
dxAPDajX6/V/I6N6bQx0jHgp
+sQslsusf5qsfSiIctoUNOT0+oN90THuK6yD6DNcERusfxoW2+5DD2sfgPr9/mDw
+OXkK+wM8Vb0eam6Ny8sxnkAUBugIQee3mAw6I9EM/
bPIG7R0wOien00MFp0YAw5XfRzr9vrPRr0wdnd3X016RmEh6IH9ccB0ZeUM1qvv7s53
9YmgiCMx5S
+KP4RFUREBKFaFSqiSLWCVVSwBdEKIr5QP6IgtYrYiNe7unpudleWpHIrxdRv4cwOm
yYVmSudg6XzCX7MPnN5sjsz+P0BoCXIz+DAsizPikB0T
+avNEP0mvwP8dD3Bw5smWgQBzJUFCFHV5jLBiGiN+R/IPD3Ar/
PpGcssCz7nCNPodQGXTYw1zEiRG/j95n0PBB00DPyKOAp9QZdx/
A2XyBH5H9IPwCEcg4FFs9LlIWtaqwfMIZ4g0TvSc8BBQbeiniE/GeSnhNDmotylJ09E
+UEBYX0BzUfAQQGhfABIK21OyV0L80QfyP+8h1hGnhbVF

+AUOfJobZ6NSV6c80Tkf0jP+H2m+qIEKeAxVyWeFpj4uUXUXQk/YLG
+SM5ZEWhK5DHHG/+Jx4JEv4ZytEL+B3FIPwWEWqPzMuaR8PMUT9T9tBR/
UGO5F0E+CnisPS/zgMcTdbeIVsn/oH5Az3mQDyWobV8cEX/i5InWV1sD/
WR4XsRjMEE3+UfgWojI/
2SxvEJ9tYHHHZN7JueJ1iNQ7yv5n3BeOeGUxgLPrRoaCXii3rfof0J
+FMonCNq4E7KtFjwR8Qz5nyhnAzTO3ZVvRuGJel9G/
A8dmGnjgbnTDSZuyBNhfkb8T6mNBhxnnb8u39DEE33/x/9oAwFE/vn+Bhvn/
kjH73y7/9EWcaz3Z+tpiuOiX2zzP8biiXn/sCnfNliRaMT/AJBzwPPypHxjZVUiNex/
qL6cvyTfelqdqBzyP6Sfzv1mXc25lYj0wP+0rYPwTw/W1r5cjchE/wM8oOhXZ
+pr8K5OpCk9cF7+co0t8JWJgn5MILq2L4EhgT7yBBx/eyyJMYq+Czw/OgvjiQya9G3Iz
+JEMqM4/Y71ncXDCQ0rbXb8wkRS41ybd8bTH3hLfyQw/aHJusZKDyU/
eLsHRpMpSVO7H96easqOt8/ujmf2wF5fAJDgioQEl0jQmo0deYAnF46mv4gk/
VUtKS2zkV/3Mzk9Mzcf1/3Mz81MTwqt+/kLc5W5R5JoGz0AAAAASUVORK5CYII=)
2x); }
</style>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.captive-portal .icon { background-image: -webkit-image-set(
url(data:image/ png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAQAAAD/
5HvMAAAEyEIEQVR4Xu3afWhVZRwH8LvdmjVSjKyQbKazia1k1sL6YzXoZdy28/
J8f5XRC7hwEVk0hZQCXY2SipCgYgappXMQ5Ya9ELZqoEM3VHqxBkapm21jOBWmb
Lfudk+0Peeu2+mc3zzk9514WbN9/zz18ds75Pc/vPM+JTP9NZzrT
+b9Fn2nGxHrainb8ggEaRgLn0IOjaMFroloU/
Ssg1yxBLZpxBD00QufpODrRiFX6IiUKXY8XcRAJsryC3nRL7EpsQJ/r0V3iSS0/
MKU4D6uoQwLY/P1q4i0aYX9xVqyPRH1jbr8UtfSbE8CDqIK60wBjOEb7aDf24CBO/
gN12LjJF4dM
+dPAoNgMdNkUxLENQltAFViJdbTWfFTcaRbiMXyCpE1CM4vR5uAzdwB/
hcxCDI5jXqGFeAadNJoGH0ErVhg3YPs46Ad9JldNS3GCAbDPkFGG/
eJGWktn3PA4IUB3oxPzGY4AXSBLCSTLgfZxx2MLV1Q5tFHeWxWQDNalHuIWeo4q9
GW4VVSSng5J0KgZ8+bkYpcEqIJk6H2y0CiW4H58gGMYQhyn8DIqUUltZIk1XF01kKU
W5yhm3gVB3Q76MDaRyXBQR1aIIJnyi6jNZUUgs8uTQfUiGDJIxZ9OPfxEwhAO0F70To5
NR5skxrsUgWaGDZLQCHBHVxnKxWjwuivRS7McK7na1kpUZkEwUzXbFiZfZcVk8pE6
xZ3u34AV5TFUsV2ObBHU0+Q5wu0eJ21krw
+eD4lzBvOYf7wLeyI5DEfLx2l1DPqoSXK8Sltis/gm41mlago/
9P2UApklZE0pkHhdAv6YliB8NwEQL9HvUwCkzZHz15g5Gw8jmR0QtsizvOoACUjAIdl
OZQUkHpCgDqd1gwRslhX3djZAmDsBwpATJPtDUZPqGFuyUWU453IWu7fFHZOvhjiQ
+aqiDheQ/bJjXpf2oP+caRA
+cgPJpkxckeZfiIHMgmirC8gee4rzo0qF6KV3IJEi86R8kg0oazQ6Iv2Uyoiajt4x/
qJ1BfcCe6JQZC/BQ82XvDLYHJPUEK3tmYHTmlotpbzCSwsDomDpc
+mD6VgHETB3M5Or2c5xUBvGTq7P98DjBEjqrBOLbD2eDZj7o+aJdhnhooJxUbd/r1cJ
+yI1mJMMBGWUSP6zlezT5SlirGNKacEB4bwKEXcxrEN7gThUGSCtILaZXMC+KiGN
+5kHYITndkSj7Ko3GTIPMEvtJFKt9LDYgiZtVQTLnqUPUOJcV8KWE95df4m85ps0+CQ
/ig68xN72lscF4yveCFTa5g9AXnDR5lczC1Jr
+4UhuCsAv6YlH3EDURFbgPCG3Y2altmPG9FIJ8LnoOYLbXNDzgi
+Myvkqii9sIuocAHZZuN9tAKi6hnaiBwka9f942y2rzFcut4tZOO

+3r5J6xcVmYEeKc9xlPuC3FhDHynBA6LQ5GGA2ftnNl82RqDrlDgb1ZU4AnxzUTc7raD
eWhwPCUW0BA/C7gYeP9cWqILSUX8YAgmxxIoF3tYL/DkK9Y/znwm8CI4lvRLV
+dVAQThOpOPht8p+ogZ4W9/jijFFD5eXhWPgPCXjMh/
pSVYD6pxaywPGOKOIBWfkYhXYKozgvy5/
rYBva8avzcx19sSrgT2jJnjJMVVMiUAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAQAAABNTyozAAAKSUlE
QVR4Xu2de2wUxx3Hfxhj5dkSlgOhpICDWilH1AjTGEwdnaCNdbi+2/n
+EGlKAvSBopRIKa+KSC5tCU
+ZoKaKQ9WSNihqgsXDqRMkHBRBDDi04Q9kGQFG4JAEqF1TIkwSP
+BcjBXK7zC3551Ze3zdz/7NnffD3uz8HjND/
hAQEBAQEBAQEBAQEBAQEOAMjeWpOVjF5djFB7gWDWjmNm5DMxq4lg9gF
5djlZoTy3OGkmceuzNWoBaiDDu5Dmf4M26/en2GM1yHnShTC2MFj91JNhG
+y4lwKVejkTtTv9DI1VzqRMJ3UaoMxlS1Bof5iuunX8FhtQZTaTD1J6EsNU2tRA06uNP7
hQ7UkCuxB3kDN/X605t4Q+xB6g8wGWV8njvNXJQUnoH9Op+O/TyD
+g5nLEpwXAjwURCK+JCR7ziEIvKf6CRUIC4E+CiIc7DT5LdgJ+eQf8QKuEoI8FdQBi/
jL41/05e8jDLIPM50rhYCfBaE+/Cei4DL+Ce2ogy/4We4i2fwG5RhK3/
Il5NLwnu4j0yC0dgqBfgvKKmeI/wSopGv0S34/tcRxR/
4SDJFZIpwplrCLdzZ54LG9DinalG/Lx5H1+AQP4dNqOSDfIovXb1O8UFUYhM/x6GvRjB
+iS/1NP/
CGDIDHuU6DQFaY5CaglZxW2d5WdE9RMV3qDn8Op9LKvwcv67mTLmdqOgePl+zQk
+rmmLo2eG1Ou8r/beY8+R1AW14PpRFFMlWv0NzqtLxb/
w2kk0UysLz3PaVlOdJMkHsfo2pmbHXPFZ1z2A4RPTDb/BG/qLX6r/
AK9FRRBzqnk1hFZkAReL/qd8E0SC8iZJwJhGewgXP+v/T9dSEM1GCN2kQaZOBdTo/
LfOhhhqOCt3Pxw413FD4iS1CgAWCsNvEN2A36RO9W/wxlggq/uYt0yjH
+DW1Ri1Us69eC9Ua/JWPCS1mX
+1quAgLrRFEFMvnNnG7cbyDeRh9i2ntPLyTMEi0YSrpgjGo5057BEkw74ab/
Ut3hieSjZn8MvbyEW7idm7il9jLL2NmJPta9ujV/
0lVPzHw9Ag9VgiScGl3Zsd5gCh3iPMEasRTlp+vGvWjcKbzAPZ16cF6A2OP
+HFZKYgyeLtaQhmUwc/ypylI/5SfDWdiMW+nDO03lxiaLRMk4RA
+SF07PuAQaZMhXuyWC8KOXqrfTrpgHXfaLUgWfHo1GBzSLv
+gCHHbBUlmjMRHqenBRzNGaoeklluayWJAsAN0Ql7VjL1bgaWcWnsYLeJ/br
+u5oF3uCWeKiN1qQRln3DXHQTN
+XXQPCQqH8fJrxag2J0y68FohwHJBEjUblWo4USwfr
+Aot3ALjqKsOxUWHcFvq9kGsoWlWyPoEnmgS0SiaFRGR5jKF9bZ8/
zwKfIA3u9JNvaSCXipRXo6sYM8oCb0kJK/
pCYYKeRwiw1idINJPH6T6sfJBNhmkx6cLb6DPKJeFKJfNFUl7bTpUj8l7wzmPdcF7THU
DcTVNunBJtliko1PuvTgk0i2oRYEm/Twn3KHkCbOI2hFq/MImYGrbJGDj/EUGUHNV/
ON9fdYoKYF9diCH4dvl/
tAhQWhg704YxEPBCUBJckFBlLqA0FJwOTkAgJBZYGgpMUdPh8ISoKallxAIGilbM7Hv
kCQADVCwKLCYXwsEHTDgiW5Isd5+Nq08V/pJ8h5WK4kSnGxlRMR/
6y5u1fPyeVL6SalBslanxOhVOBSIWjbDQuNOqwWpJ8vLfWQJFML6Dr4eboJUguEoGp
KBdnlJzP/WGFaDy70q6AJsjsxpaW2cuBKzOLxa4YFvU/
9SO4QOWyksHw4licE1ffwke8aFNSG79oUlMfyyA01V9zA2z02AB82lucc/
s4h6mdkSVrNITewWjxB66kHoqPwcboEElgv7ne1u9FyYXT+Lfv/
LqSHIDVf3EF5bysZhcn6btJBEBcKQVW9jMOSLShznkA8DZ6gKWJIqXE3WitG9YcoCfj

VwBcUe0jcQS25gQYRnYzVzTzaX70RT1ADuSHDN3Wva0/7WwNbkLo3ITR3Qw69oSx
yYcrtfHAgCwplyYmrIUGyZ4JPpJkg/Z+YJDqem9LvJ6Y/SMsl7os0G6T1X/MSFeMr/
xeveT4gBOVTyuAXPheIzmEr5xgXlC8EHSA3sEuGGpb147eob/kZamCXTrDqziB
+w29FqPQ3WNVNd7i/
NPf4rKjJ13THKnejc1wSZq4pW78XLvibMPOQcvWwsuyMrYLcU64aSfvUUd/
BxYEhKHcIX5ZJey9lHw+oH3C7HYL0yz7uhUNPqLlWCNIvHLqXnm1rA/Wz9KzRvOAF/
NkCQfrNC+7tLxpLgXdalEi3/
cW9gYo8IBb3WyqIF8mEvdcWvH2kQXQEGmwVJFsL1UrvTZw5Woq
+zedtFMQ5Urya5r0NeDlpob7HX1ooaLkQdD6U5b2R/
ARpgpl8xTZBMpOOMq2lCLF80gS/
tEtQLF9qx2TqHXIfcWwmbewShM1Cz3HNeTA6iselk6DicXKuhxL9BXUb00kQ/
ij0xJ2x2ksy0RodlS6CoqPQKgRVmFnUuyFdBPEGqTw6yciycHyOMekgCGMSipxV5jYW
eIs08G3h+HF+w5mV+jQvsR8lVmBwawoU2SNIXjjJTCmAogTB1WY3NznIfYm//
6sYsd5tJ83wbXxKCnKmkx6Jx0BgBXkE//
BbkHsdDy9IPdhqfIMldHg4KUCkcn2+OFnDJjrk
+IXRpI9aIq3z6cJh5AEnty8E4aQoVMltAU9LmWqJ0U3e9OvjeLcvFDmzqEdQmSCzLpx
JAoPbBGKxtyoUWv0XhL9RD2Cx1lM4HiWB2Y0m272N/
mp2Hwg63uPbOLGQuZYExrcqxUVMJA/gZ9zudw8RJYCJiaVw7Hf9eelvdovG6HiP
+4J86K+ixBZTNEo9aI7dT+7ob5eMkx43GR7kRPhVHMVFvsKX/
RU0Y2Ti1BBxl3jA6IbbtUKRJSGI0FObqA/ryAWjW7bjZHS8rYKi43HyJj1bXMIR85v
+oxETbRSEiWi8Sc9ut6jfl2MjcNGZbpsgZ3oPTVyHoneTBjoHj7TzIpsE8aKbJxOo1zr1Sf/
oGlQWDrND0PWgQujRyIcaO/zotJqiIcjP65DG02Py+Cx0YEX4NtsEYbfO2GP
+ALZTKLJJELbov7nMH+H3FsbYIAhxrNOY9/h5CCQ+5w3RUf0rCM26QYXPx4iilTcWj
+svQdhvIiT1/SBadGBzLL+vBSHOa00mNPw/yvgEL
+ecPhNUJ7OF9jxHS5MfMYF9vMh5mAb5KqiFl4YzyVYwGttSGDq3qQVqQu4Q84KwDa
PJDvQP5EcH6vltrDemp1qEzHYTK+jjAwOqYgU00IhOQoX/
274jjgrR3zPgdsYvQb1vcupR4oylgQ8mo8xwb8d5lGEypROhLDVNrUSN3nan6ECNWq
mmhbIoXQnf5US4lKvR2CsxjVzNpU5ELFhKd5yhsTw1F6u5nKtQw7VoQDO3cRua0cC1
qOEqLsdqNTeW5wwla/gvpXzJeo7GTncAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .captive-portal .icon {
filter: invert(1); }
} </style>
.https-only #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.https-only .icon {
background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/glDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzXXRD
XXQzbXQzbWQjXYSDvWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP

RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLIPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////
PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTIMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnUFkRXwLiK78llgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPIEITpIRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3DiHUhftKIIzdKoRSFe0r8sXDAkSoumkIigYaIOkIf
eWi56EESFm8r1w0fFll4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRl/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXiz0987BBtux65qaCeF8lHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7IcfH9E3ZlEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9rkJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKFtf+huwrjaa6Pxv7RfgW7nubJPtKl/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAEIFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAJAAAACQCAMAAADQmBKKAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTIMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56ISgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vvRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMIL
5W6t4Zkm22lkm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjITvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYI0gGFkvaR779KKZUWuFKe7nIIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGI8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7ZZl
+ShNVMrdW2Wg/v

+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjlMbDK01U0OVZxcplE6BljLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyleiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFl83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiljRDx
KQwJk9a+o2u431gYZERBo/kcBlfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzIHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPl
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
filter: grayscale(1); }
@media (prefers-color-scheme: dark) { .https-only .icon {
filter: invert(1) grayscale(1); }
} </style>
.insecure-form #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.insecure-form .icon { background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAYAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqWxeWyLjRH60BYpKZHYpoFC
U60/
xKCt5ME3QaSpT6WUPElCEXyTUplojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2
WZPefO7LjrzjYzcJmZc8495zvf3Ll3Zu
+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7D7ivo1Ssrm5WYL9AZSC7OzsAuyzl
HuCHCsjyOawZ7lbVFT0W09Pzz843Pz843rNtTwhqaGh4ZXV1tQFZfYSDgKe85MhyFpy
V/Py8g
+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGovr6+MlxBp
kgOuILVeKVJnJzc78eGBi4nXXi4GBi4nXXrMKXvY1GjRmvrz8/Pb+/
r65pMVlWkkEodV8vLGx8SPI2Z8scJ78gHVrMKXvY1GjRmvrz8/Pb+/r65pMVlWkkEod
DOkcQybG9ifwv6OezvRyKKRv5eWljhyZeG4AMcvweYNnnHKkq4TNcezzqXfbYLsBm46
oaELbrZu+l0R1Nra+vz8/HwwPgPGPGH/uKgj6xwA
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4t1truru7/1VMXMW
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4t1turu7/1VMXMW

+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqIPo975FrGbEytV
8QT
+JlnTMT0vyRRD6nEsAZLutOIpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXFf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8In1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYlaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xlpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVIGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6NoI
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+lVbEHR3pcBg
+zgXEki6IMuImdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusCIDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwIjMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnElXSzlth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYIhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yIuhn8Aw+70ZTSym/lX0zRiMxZiaJ2IlZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz
+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh
+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWlBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAJAAAACQCAYAAADnRuK4AAAAAXN
SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxSZTsw25wAUPiNQTRgFkv
8YIbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39alT5+m5rCEhAICAQEAgIBgYBAICAQEAgIBgYBAICAQEA
zPv6urq6q7m5rCEhAICAQEAgIBgYBAYBAICAQEAgIBgYBAICAQEA

gIBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTPryeeeaZX65fv/5/HI
+pZUwQ6I477vjD0NDQAgiwgOBfynYa23E+I43OY+jcy/Zjtn0tLS19zz///Oc
+y8ijroYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGQF1dXRP2799/
G2TpBLCbWFuyBM5B9xB5XoVIPVOnTn2xu7v7slOO3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPiIvWRWrcEWrx48XyI8xA1znX1AXVxK6mR3oBI
qzdv3qxbXd0tdUcgapybIY2IM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtbLcoeiR4
Mg0UesXRDpPY9qM1OVewItW7asjT6bJ0DgL6y5t9dTpI6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4lUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUMa+2o6/fTTJ//yyy+Tjx49OjxOhsxFJA+PobE/
PJ5G3kmSrcLyZFtb2wNr1qw5UoWyKhaRCwItWbLkIsaqthCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFjjupOyYaecbt3756Nbo21acztGraZEQr97zPW1vHcc899dYohN
TioOYFo78ygvfMavl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp2bNnzzz86oRI7RSo0X3fyz78uoF20R7fii36akqgqG/
nZUA+12J0JVII8zrr08htA+BDIeSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqlfQuwgy
+XqA+ZmaaDEkqkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/
JEpCFItBwSqeujqkvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26nAvgKXy43INKkej2atdEvqrRRP6rzRPdtlKRB9APANa
9s2bNuqpeySPAZLt8kC/yKRGIpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axAnzUsJ5HTVSVp7OqEEj9PNzz3wWYNI/
qqqlfZt7MEwCUy3GhNIFXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cl9amSvkzv4Wph1mdhBiShjzq85jPVfV4o5JHgZJv8lG
+cpgm+BcePny4V9hLb5ZL5gTS8ARXVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWpsE/+j1edPWAK/It1oUA
+qOrjnrOR7vxLIiwnfVaVz/oF7uN2/5Lrvkvkr5cusBsL5adzL11cyoNR5iLNt0qRJN45V8ggX
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQIwFg7xuKbnqMDKQyEhSuJFIMoFpncbTIhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4YzI1i4pi9dDbvrIzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjElwe8eOXE0b0+djxWgNxhWio4h0Ms
+pVJ6H6eWr2qM64IKIzkmEIq48+4jWsA5yvBuedHLQYlR4H57ng7O2VIa81EA22bhwy
A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhIxYspnUrDcIeGJ0pLhDPFEIi
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AlDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC
WnIEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqeYbGoCYcjKYoorpD1Tzz
MJK/hW9dMRls9YC3aM3SnFpcKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qkIhJplJ1UI29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/

Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujIW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc
9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0IhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6IutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrIyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMIOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYlb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMILmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNlsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnIZDpNkahGpX/
U2HhYT8TBISxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7Ie
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqlMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTIg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAleC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9EIg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UIgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xlvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrISZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY
+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYlPK49jRrMHC7yyf3vaKSQ2XCR
Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGlYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh
+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1IY9SS3ru4U
+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxIkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/

DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFlGWkt5aeM64/
zv+MLwSXf/
lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0XxfrrdQtRNaq
+lsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOIctHrF217Nux4Fk3BIqdgkg3Q6KHWF0nqc
WqcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tIME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodIQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .insecure-form .icon {
filter: invert(1); }
} </style>
<style>/* Copyright 2019 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.lookalike-url #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.lookalike-url .icon {
background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAYAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqWxeWyLjRH60BYpKZHYpoFC
U60/
xKCt5ME3QaSpT6WUPElCEXyTUpIojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2
WZPefO7LjzjYzjYcZ8c8495zvf3Ll3Zu
+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7ivo1Ssrm5WYL9AZC7OzsAuyzl
HuCHcsjyOawZ7lbVFT0W09Pzz843Pz843rNtwhqaGh4LxKJ/Py843rNtwhqaGhoqL
V/Py8g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5g5gEcIvB/Q079+h6
myXwAAAAASUVORK5CYII=

+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGovr7+2Pr6+j
kgOuILVeKVJnJzc78eGBi4nXhVe42kEtTY2Fi8vLz8HVrMKXvY1GjRmvrz8/Pb+/
r65pMVIWkEodV8vLGx8SPI2Z8scH78gKTFnJyc02hN1/3Ud9ZJCkG1tbVfwnEnyMIxBp
DOkcQybG9ifwv6OezvRyKRv5eWljhyZeG4AMcvweYNnHKkq4TNcezzqXfbYLsBm46h
oaELbrZu+l0R1Nra+vz8/HwPgH/uFgj6xwA
+inlNt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4tburu7/1VMXMW
+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqlPo975FrGbEytV
8QT
+JlnTMT0vyRRD6nEsAZLutOIpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXRXf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8In1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYlaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xIpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVIGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6NoI
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+lVbEHR3pcBg
+zgXEki6IMuImdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusCIDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwIjMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnEIXSzIth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYIhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yIuhn8Aw+70ZTSym/lX0zRiMxZiaJ2llZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz
+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh
+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWIBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAElFTkSuQmCC) 1x,

url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAYAAADnRuK4AAAAAXN
SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxQSZTsw25wAUPiNQTRgFkv
8YIbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENFZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39alT5/
zPv6urq6q7m5rCEhAICAQEAgIBgYBAQCAgEBAICAQEAgIBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423Hjxvj423HjxvD4+3HjxxWTPryeuraq6q7m5rCEhAICAQEAgIBgYBAAg
IBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTPryeeeaaZX65fv/5/HI
+pZUwQ6l477vjD0NDQAgiwgOBfynYa23E+l43OY+jcy/Zjtn0tLS19zz///Oc
+y8ijroYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGQF1dXRP2799/
G2TpBLCbWFuyBM5B9xB5XoVlPVOnTn2xu7v7slOO3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPiIvWRWrcEWrx48XyI8xA1znX1AXVxK6mR3oBl
qzdv3qxbXd0tdUcgapybIY2lM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtLcoeiR4
Mg0UesXRDpPY9qM1OVewItW7asjT6bJ0DgL6y5t9d9Tpl6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4lUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUUMa+2o6/fTTJ//yyy+Tjx49OjxOhsxFJA+PobE/
PJ5G3kmSrcLyZFtb2wNr1qw5UoWyKhaRCwtWbLkIsaqthCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFFFjjupOyYaecbt3756NbbBno21acztGraGaZEQr97zPW1vHcc899dYoN
TioOYFo78ygvfMvl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp2bNzzzz86oRl7RSo0X3fyz78uoF20R7fii36akqgqG/
nZUA+12J0JVll8zrr08htA+BDleSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZJM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqIfQuwgy
+XqA+ZmaaDEkqkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/dKLjxw58iC67/
JEpCFtBwSqeujqvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26atdEvqrRRP6rzRPdtlKRB9APANa
9s2bNuqpeySPSPAZLt8kC/yKRGlpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axAnzUsJ5HTVz3wWYNI/
qqqlfZt7MEwCUy3GhNIFXXsjTTG/z/
dQkj3KYpybeN3HixDbDixKYpppbeN3HixDbDixZkbN27cl9amSvjTG/z/
dQkj3KYpybeN3HixDbDixZkbN27cl9amSvjTG/z/
dQkj3KYpybeN3HixDbDixZkbN27cl9amSvjTG/z/
dQkj3KYppbeN3HixDkbN27cl9amSvkzv4Wph1mdhBiShjzq85jPVfV4o5JHgZJv8lG
+cpgm+BcePny4V9hLb5ZL5gTS8ARXVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWXVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWXVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWpsE/+j1edPWAK/lt1oUA
+qOrjnrOR7vxLliwnfVaVz/oF7uN2/5Lrvkkr5cusBsL5adzL11cyoNR5iLNt0qRJN45V8ggX
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQlwFg7xuKbnqMDKQyEhSuJFIMoFpncbTlhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4Yzl1i4pi9dDbvrlzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjEIwe8eOXE0b0+djxWgNxhWio4h0Ms
+pVJ6H6eWr2qM64lKlzkmEIq48+4jWsA5yvBuedHLQYIR4H57ng7O2VIa81EA22bhwy
A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhlxYspnUrDcIeGJ0pLhDPFEIi
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AIDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC

WnIEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqeYbGoCYcjKYoorpD1Tzz
MJK/hW9dMRls9YC3aM3SnFpCKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qklhJplJ1Ul29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/
Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujlW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc
9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0lhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6IutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrlyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMIOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYlb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMILmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNlsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnlZDpNkahGpX/
U2HhYT8TBlSxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7Ie
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqlMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTlg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAIeC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9Elg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UlgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xlvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrISZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY
+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYIPK49jRrMHC7yyf3vaKSQ2XCR
Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGlYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh

+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1IY9SS3ru4U
+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxIkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/
DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFlGWkt5aeM64/
zv+MLwSXf/
lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0XxfrrdQtRNaq
+IsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOlctHrF217Nux4Fk3BIqdgkg3Q6KHWF0nqc
WqcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tIME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodIQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .lookalike-url .icon {
filter: invert(1); }
} </style>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
body.safe-browsing {
--google-red-900: rgb(165, 14, 14); --google-red-700: rgb(197, 34, 31); --google-
red-600: rgb(217, 48, 37); --google-red-500: rgb(234, 67, 53); --google-red-100:
rgb(250, 210, 207); --google-red-50: rgb(252, 232, 230); background-color: var(--
google-red-600); color: white;
}
.safe-browsing :-webkit-any(

```css
a, #details, #details-button, #proceed-button, h1, h2, p, .small-link) {
color: white; }
.safe-browsing button { background: white; border-color: white;
color: var(--google-red-700);
}
.safe-browsing button:active {
box-shadow: 0 1px 2px 0 rgba(165, 14, 14, .3),
0 2px 6px 2px rgba(165, 14, 14, .15); }
.safe-browsing button:hover { background: var(--google-red-50);
}
.safe-browsing .secondary-button { background-color: var(--google-red-600); border-
color: var(--google-red-100); color: white;
}
.safe-browsing .secondary-button:active { box-shadow: 0 2px 3px rgba(0, 0, 0, .5);
}
.safe-browsing .secondary-button:hover { background-color: var(--google-red-500);
border-color: var(--google-red-50);
}
.safe-browsing .error-code { display: none;
}
.safe-browsing .icon { background-image: -webkit-image-set(
```

url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAMAAABiM0N1AAAA
+VBMVEUAAAD///
////////////////////9/f3////9/f36+vr8/Pz7+/v7+/v7+/v7+/v////
r6+vn5+fk5OT5+fn19fX4+Pju7u7v7+/o6Ojx8fH09PTy8vLz8/j4+P39/fm5ubp6eni4uL8/
Pzw8PDt7e329vbs7Ozg4ODh4eHe3t7l5eX6+vrd3d3q6urf39/c3NzbRDf7+/
vb29vW1tbZ2dk+D9arAAAALXRSTlMA/
eIxBfACHj3YwhYJDyfOtfr2WHObgEqpjellgY6c6mZLV3L2qKjOZemN
+rUv7NpAAAACcUIEQVR4Xu2U53baQBQGAxiMaQZT3Wt6rnql9uKW9v4Pk5XEspG4
WFmJ/
GMeYM635cy7/8yePXv23N7uxnNwdHSwE1ENoLYLTz4NkM7vQHQFhKv4nlQCCIIUbF
EOXHJxPVVYUY3nKSSpKFmIJbqANRdxPIcnTHRyGEPUAga0onuy4CMb1ZM5Bh/
HmYiiGwhwE81TvIYA18VIokvY4DKKp1SBDSqlCKKyIJT5PU1AafL3FVC4q1uDLdS4+4r
CXd06bKXO21fKdxcNKFzVzUFQ9KoDJRehr0z0tBgBpcrfVyZavAxM7uo2ABHNO8s2N
TW4+8pEs4HUMxS+6rYAEZFBYl+Y8FQ3C5hoIImGMNVVVvLp4X1FRZ2n0BU035X
+u7jmgomWv/zzWR23lB7ich/
b1DBeRQVNNty1FHYLDWVh1TwEXic4gs6105UfPdMrd1/
HCffrnqTayrYkqP3qmSom7r9q84xyMDLIUx7Mylfn7OvlGmbYyUYmGbmry99UWnYOZ7
sFc5OHb1b2HLbS9QV1vED3dPX9fQXFuWiEeF9kzpfMcfaUZkbyn9+bQTfXwvm4W0iA
3RNdQUyIV1lekkDOhSx3sdGh17+At0ctA0plDpr/gLqSvSI
+kXt8mAj9DVl20r3hGBM0iOwKmBtZXHOPX79fFXCJ/
Ure79Gxr40Owuu9hK8LPp1mnR7JmWuv7ZsM+oH3FGZObFoWx3iY/
Ul1J2LVn0b7i6B2RZZM1WumwL833MoH3FMQ1hPHIGsadnvk94X3EsctPE4xK8p89FvK
84ExIRuoUa6Pt98fU1DJUMosj+3/S15OtrGOsB9LnYqm+e5w
+es4JAhYQDdgAAAABJRU5ErkJggg==) 1x,

url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABPl
BMVEUAAAD//////////////////////////6+vr7+/v7+/v7+/v9/f36+vr7+/v7+/
v///9/
f3/////////8/Pz////5+fn/////////6+vr/////////8/Pz7+/v7+/v////k5OTh4eHf39/
e3t7g4OD6+vrbRDf5+fnv7+/x8fH09PTY2Njc3Nz39/fq6ur4+Pjz8/Pn5+f7+/
vZ2dnW1tbt7e3U1NTV1dXl5eXr6+v19fXy8vLs7Oz29vba2trb29vo6Ojm5ubu7u7d3d38/
Pzp6enj4+Pi4uLw8PD///
+BQ30nAAAAQHRSTlMAHhYFDzEJAifiwvb98M76PdjqtoGpZll0c46qj1icTJ1KS42A6W
Wb2WdadYK1cvCb/cJXV+KA9qioPc5l6Y36uRjySgAABXVJREFUeF7s1kuKwlAUBNDS
+JtqggkkkmODIH4gPRIgD11P730BDI6HV260Z1Wu4ZxUH/
4tzzjnnnHPOOeecq2tEJdlsEsRkT
+4RkeGRPA4Rj4YkG0RjfCLJ0xixCPwWEIkd73aIwuTMu/
MEMTiwc4AeBhd2LgPolfyhhFzLBy3Eplc
+uE6hteWTLaRGCz5ZjKBU8UUFodmNL24z6KQ0pJCpaaohksxpmifQKGhjIXurSbbZhr
9qRG81yTYbaJJtNuefcsFbTbLNZnwjE7zVINtsybdK4Vttreyt Hdlml/
zIUvTWjmyzFT9USd7akW02pUm22TV7WOveKttswV4K1Vtlm12xp5Xmrb
+15YK3qjab0STb7Fcv5tmcOBKE4Q2Xc7KrTJW3yhs/
2RsuZwvBjsQShPd0JxCyJVmA5v//getpgYextLSYKdHf
+PbUO93No36Ht77eqHCvNnvASaBLp4rooCIvpYGurpx92Sx6Kw3Uv873ZbOHvA7Qtd
12m7ZZ6a00UN
+225a7F5tt8XpA7Y61rJi1ViPeSgPZ7XZnuVh4zdsseisNhAEtzgOvaZt9yGsCdSwLeJKk
16DNSm
+lgSCghQAax9NGbfaI1wQSD7YIgnE8H0a3iiY4a8FYSyMKAxgJoMqBs1txbaSDs6AR5
Zv6Asllzb6WBIKAkiefzCQA5A8Jmjb2VBtoMKHXcUSM2e8x3ADoPAAh4Jr7v5G7IVKLjh
r21DHS
+DshP0xyAel3CZk29lQa6ebAUeLxe74KwWTNvpYGSFVDqOK4LPFF0QdisibfSQBsdj
QFNo9EoI2zWxFtpINiJw+Fs5hcBTaeDwaibETar76000GZAnggIeLqSyNxmD/
luQOM4LngceDBP8ACQSnTNY+L111geRtqxlMfLDAsiFkV8H1GWSyNBmW7xO
+eiJyo7O87B4MCaA1lxazd9bUwgIgFYd7a8Dwo4GIqYQgc02f291kEcd
+YJHVmZusye8djkAIKxHHnjEiEUIVOTDFKKTPdxbXdnRjghI7GjgUSqTNmvurTRRkMQ
AtBp53IkjJuNRiY72cW8NiwdLHdyJgmcdkGTKTGz2Gd
+xPOhoGVAUjZgCo8zas33cW3lviFqWhyF2ND7YbaZM2qy5t9JEs8ITPexoJlFksUzTZo
+5Tk3FiIWh7KAKpEzarLm30v/2ckczlYXdsGU6NvuY6wFdJjn+y4OXKR3EytP/
WMNbNYD6Y9Ag8WCs/F4yo51t9oBrAoGJzJV/
+WokJDow9NbaF7QJAKKkNpMyb9KPfjL2VBrLtznLpQ0DlYNTKdrHZp1wfqA0iEjhMYsisl
N9I9LSmtz7RBxLmGCSxW9k9pYye1LPZR9wACNQagIbutgaSfvTI3Fvp
+xDwgDd6ZERAVM9mT7kB0HIZJIn4Ekp7W+KRRKfG3kqfYyAg4HHy6ZZwpB
+daHgrVfG///
3z5u1l38aODsbFrcH1opJ7VPgRabNnfPeaI8+13REdPV4FBA4CRkQyZWekt2rU8M3lV
REQns9AG4XmgxLRjc0+v0d4q1ZNVjznNw8WgsTiviaRXmh5K/19jzsaL
+biwVwhsbeXdLUfffa+nrfSX9P4rYjnRcdFBWGSQ+KUf3/
ybp7vuX6lFp7woaN9eDBxiUEIwCDH/
wfCWzUrXwTrgLCj14rGuhTSxx8Q3qpZLnSQ4MlXHS2He8u
+xvqJ8FbdCmEFTXxcQdFAJkPP2qfVNvsLNy1vjg9WBKQsQMKPfiW8Vbt6SkdLFkb50
VeEt2rX1MGA4MHUlhkRfvTFh4S3alcEHaSczmr20R+Et+rXCDu63DJsux/

```css
9+R7hrQZE0NFEHlVh/UV4q0F1B6zSEdlWF/lSsdn/
AccLgKctjBheAAAAAElFTkSuQmCC) 2x);
}
@media (min-width: 240px) and (max-width: 420px) and (min-height: 401px),
(min-width: 421px) and (min-height: 240px) and (max-height: 560px) {
body.safe-browsing .nav-wrapper {
background: var(--google-red-600);
box-shadow: 0 -12px 24px var(--google-red-600);
} }
.safe-browsing-billing .icon { background-image: -webkit-image-set(
url(data:image/
```
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/glDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbcRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzXXRD
XXQzbXQzbWQjXYSDvWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP
RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLlPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////
PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTlMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnUFkRXwLiK78llgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPIEITpIRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3e3DiHUhftKllzdKoRSFe0r8sXDAkSoumkIigYaIOklf
eWi56EESFm8r1w0fFll4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRl/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXiz0987BBtux65qaCeF8lHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7IcfH9E3ZlEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9rkJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKftf+huwrjaa6Pxv7RfgW7nubJPtKI/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
glDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P

WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56lSgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMlL
5W6t4Zkm22lkm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjlTvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYI0gGFkvaR779KKZUWuFKe7nIIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGI8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjlMbDK01U0OVZxcplE6BIjLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyIeiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFI83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiljRDx
KQwJk9a+o2u431gYZERBo/kcBIfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzIHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoiIG6ZKuBH184D
3qpdAQIIWp0p9dE7wFv1q8Y6+njLoPI
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
}
.safe-browsing-enhanced-protection-message {
background-color: rgba(0, 0, 0, 0.2); /* black 20% opacity */
border: 1px solid rgba(0, 0, 0, 0.3); /* black 30% opacity */ }
.safe-browsing-enhanced-protection-message .icon { background-image:
url(data:image/svg
+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdmciIHdp
ZHRoPSIyNCIgaGVpZ2h0PSIyNCIgZmlsbD0iI0ZGRiI
+PHBhdGggZD0iTTkgMjFjMCAuNTUuNDUgMSAxIDFoNGMuNTUgMCAxLS40NSAxL

TF2LTFIOXYxem0zLTE5QzguMTQgMiA1IDUuMTQgNSA5YzAgMi4zOCAxLjE5IDQuN
DcgMyA1Ljc0VjE3YzAgLjU1LjQ1IDEgMSAxaDZjLjU1IDAgMS0uNDUgMS0xdi0yLjI2Yz
EuODEtMS4yNyAzLTMuMzYgMy01Ljc0IDAtMy44Ni0zLjE0LTctNy03em0yLjg1IDExLjFs
LS44NS42VjE2aC00di0yLjNsLS44NS0uNkM3LjggMTIuMTYgNyAxMC42MyA3IDlljMC0
yLjc2IDIuMjQtNSA1LTVzNSAyLjI0IDUgNWMwIDEuNjMtLjggMy4xNi0yLjE1IDQuMXoiL
z48L3N2Zz4K);
} </style>

<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the LICENSE file. */
.ssl .icon {
background-image: -webkit-image-set(
url(data:image/

png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/glDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzXXRD
XXQzbXQzbWQjXYSDvWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP
RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLIPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////
PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTIMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnUFkRXwLiK78llgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPlEITpIRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3DiHUhftKIlzdKoRSFe0r8sXDAkSoumkIigYaIOkIf
eWi56EESFm8r1w0fFII4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRI/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXiz0987BBtux65qaCeF8lHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7lcfH9E3ZlEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9rkJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKFtf+huwrjaa6Pxv7RfgW7nubJPtKI/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAAEIFTkSuQmCC) 1x,
url(data:image/

png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvcWQjbVQjXIPjLOQDPXQjbCPDDNPzzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/

MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56lSgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMlL
5W6t4Zkm22lkm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjlTvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYl0gGFkvaR779KKZUWuFKe7nlIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGI8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1ulQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjlMbDK01U0OVZxcplE6BljLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyIeiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFl83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiIjRDx
KQwJk9a+o2u431gYZERBo/kcBIfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzlHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPl
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
}
.ssl .legacy-tls.icon { background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAYAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqWxeWyLjRH60BYpKZHYpoFCF
U60/
xKCt5ME3QaSpT6WUPElCEXyTUpIojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2

WZPefO7LjrzjYzcJmZc8495zvf3Ll3Zu
+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7ivo1Ssrm5WYL9AZSC7OzsAuyzI
HuCHcsjyOawZ7lbVFT0W09Pzz843rNtTwhqaGh4ZXV1tQFZfYZSDgKe85MhyFpBvTso
V/Py8q5g
+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGovr7+2Pr6+j
kgOuILVeKVJnJzc78eGBi4nXhVe42kEtTY2Fi8vLz8HVrMKXvY1GjRmvrz8/Pb+/
r65pMVIWkEodV8vLGx8SPI2Z8scH78gKTFnJyc02hN1/3Ud9ZJCkG1tbVfwnEnyMlxBp
DOkcQybG9ifwv6OezvRyKRv5eWljhyZeG4AMcvweYNnHKkq4TNcezzqXfbYLsBm46h
oaELbrZu+l0R1Nra+vz8/HwPgH/uFgj6xwA
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4tburu7/1VMXMW
+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqIPo975FrGbEytV
8QT
+JlnTMT0vyRRD6nEsAZLutOIpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXFf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8In1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYlaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xlpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVlGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6NoI
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+IVbEHR3pcBg
+zgXEki6IMuImdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusCIDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwIjMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnElXSzlth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYIhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yIuhn8Aw+70ZTSym/lX0zRiMxZiaJ2IlZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz

+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh
+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWlBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAEIFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAYAAADnRuK4AAAAAXN
SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxQSZTsw25wAUPiNQTRgFkv
8YIbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENFZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39alT5/
zPv6urq6q7m5rCEhAICAQEAgIBgYBAQCAgEBAICAQEA
gIBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTPryeeaaZX65fv/5/HI
+pZUwQ6l477vjD0NDQAgiwgOBfynYa23E+l43OY+jcy/Zjtn0tLS19zz///Oc
+y8ijroYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGQF1dXRP2799/
G2TpBLCbWFuyBM5B9yB5XoVIPVOnTn2xu7v7slOO3O3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPiIvWRWrcEWrc48XyI8xA1znX1AXVxK6mR3oBl
qzdv3qxbXd0tdUcgapybIY2IM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtbLcoeiR4
Mg0UesXRDpPY9qM1OVewltW7asjT6bJ0DgL6y5t9dTpI6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4lUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUMa+2o6/fTTJ//yyy+Tjx49jxOhxOhsxFJA+PobE/
PJ5G3bkmSrcLyZFftb2wNr1qw5UoWyKhaRCwItWbLkIsqqththCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFjjupOyYaecbt3576Nbo21acztGraZEQr97zPW1vvHcc899dYohN
TioOYFo78ygvfMavl+Mavl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp2bNnzzz86oRI7RSo0X3fyz78uoF20R7fii36akqgqG/
nZUA+12J0JVll8zrr08htA+BDleSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqIfQuwgy
+XqA+ZmaaDEkqkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/
JEpCFtBwSqeujkvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26adeuXatdEvqej2RRdtlKRB9APANa
9s2bNuqpeySPAZLt8kC/yKRGIpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axAnzUsJ5HTVSVp7OqEEj9PNzz3wWYNI/
qqqIfZt7MEwCUy3GhNIFXXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cX27c9v9amSvkzzKEwCUy3GhNIFXXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cX27c9v9amSvkzzKEwCUy3GhNIFXXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cl9amSvkzv4Wph1mdhBiShjzq85jPVfV4o5JHgZJv8IG
+cpgm+BcePny4V9hLb5ZL5gTS8ARXVVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWpsE/+j1edPWAK/It1oUA
+qOrjnrOR7vxLIiwnfaVz/oF7uN2/5Lrvkr5cusBsL5adZL11cyoNR5iLNt0qRJN45V8ggX
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQlwFg7xuKbnqMDKQyEhSuJFIMoFpncbTlhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4YzI1i4pi9dDbvrIzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjElwe8eOXE0b0+djxWgNxhWio4h0Ms

+pVJ6H6eWr2qM64lKlzkmEIq48+4jWsA5yvBuedHLQYlR4H57ng7O2Vla81EA22bhwy
A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhlxYspnUrDcIeGJ0pLhDPFEIi
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AlDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC
WnIEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqeYbGoCYcjKYoorpD1Tzz
MJK/hW9dMRls9YC3aM3SnFpCKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qklhJplJ1Ul29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/
Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujlW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc
9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0lhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6lutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrIyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMIOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYlb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMILmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNlsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnIZDpNkahGpX/
U2HhYT8TBISxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7Ie
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqlMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTIg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAleC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9EIg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UIgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xlvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrISZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY

+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYIPK49jRrMHC7yyf3vaKSQ2XCR
Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGIYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh
+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1IY9SS3ru4U
+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxlkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/
DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFIGWkt5aeM64/
zv+MLwSXf/
lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0Xxfrrd QtRNaq
+IsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOIctHrF217Nux4Fk3BIqdgkg3Q6KHWF0nqc
WqcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tIME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodIQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwAAAAASUVORK5CYII=) 2x);
}

```css
@media (prefers-color-scheme: dark) { .ssl .legacy-tls.icon {
filter: invert(1); }
}
.ssl-opt-in .checkbox::after { border-color: #696969;
}
.ssl-opt-in .checkbox::before { border-color: #696969;
}
.ssl-enhanced-protection-message {
background-color: rgb(248, 249, 250); /* google-gray-50 */ border: 1px solid rgb(218,
```

220, 224); /* google-gray-300 */
}
@media (prefers-color-scheme: dark) { .ssl-enhanced-protection-message {
background-color: rgb(32, 33, 36); /* google-gray-900 */
border: 1px solid rgb(95, 99, 104); /* google-gray-700 */ }
}
.ssl-enhanced-protection-message .icon { background-image: url(data:image/svg

+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdmciIHdp
ZHRoPSIyNCIgaGVpZ2h0PSIyNCIgZmlsbD0iIzY5Njk2OSI
+PHBhdGggZD0iTTkgMjFjMC4uNTUuNDUgMSAxIDFoNGMuNTUgMCAxLS40NSAxL
TF2LTFIOXYxem0zLTE5QzguMTQgMiA1IDUuMTQgNSA5YzAgMi4zOCAxLjE5IDQuN
DcgMyA1Ljc3YzAuU1LjQ1IDEgMSAxaDcuNDUgMS0xdi0yYj2Yz
EuODEtMS4yNyAzLTMuMzYgMy01Ljc3IDAtMy44Ni0zLjE0LTctNy03em0yLjg1IDExLjFs
LS44NS42VjE2aC00di0yLjNsLS44NS0uNkM3LjggMTIuMTYgNyAxMC42MyA3IDljMC0
yLjc2IDIuMjQtNSA1LTVzNSAyLjI0IDUgNWMwIDEuNjMtLjggMy4xNi0yLjE1IDQuMXoiL
z48L3N2Zz4K);
}
.ssl-enhanced-protection-message a:link { color: rgb(26, 115, 232); /* google-blue-600 */
}
@media (prefers-color-scheme: dark) { .ssl-enhanced-protection-message a:link {
color: rgb(138, 180, 248); /* google-blue-300 */ }
} </style>
<script>// Copyright (c) 2013 The Chromium Authors. All rights reserved. // Use of this
source code is governed by a BSD-style license that can be // found in the LICENSE
file.
/**
* @fileoverview Assertion support. */
/**

- * Verify |condition| is truthy and return |condition| if so.

- * @template T

- * @param {T} condition A condition to check for truthiness. Note that this

- * may be used to test whether a value is defined or not, and we don't
  want

- * to force a cast to Boolean.

- * @param {string=} opt_message A message to show on failure.

- * @return {T} A non-null |condition|.

- * @closurePrimitive {asserts.truthy}

- * @suppress {reportUnknownTypes} because T is not sufficiently

constrained. */
/* #export */ function assert(condition, opt_message) { if (!condition) {
let message = 'Assertion failed'; if (opt_message) {
message = message + ': ' + opt_message; }
const error = new Error(message); const global = function() {
const thisOrSelf = this || self;
/** @type {boolean} */ thisOrSelf.traceAssertionsForTesting; return thisOrSelf;
}();
if (global.traceAssertionsForTesting) {
console.warn(error.stack); }
throw error; }
return condition;

} /**
*
*
*
*
*
*
*
*
*
*
* *}
-      * assertNotReached();

-      * return document; *}

Call this from places in the code that should never be reached.
For example, handling all the values of enum with a switch() like this:
function getValueFromEnum(enum) { switch (enum) {
case ENUM_FIRST_OF_TWO: return first
case ENUM_LAST_OF_TWO: return last;
*
* This code should only be hit in the case of serious programmer error or
* * * * */
unexpected input.
@param {string=} opt_message A message to show when this is hit. @closurePrimitive
{asserts.fail}
/* #export */ function assertNotReached(opt_message) { assert(false, opt_message ||
'Unreachable code hit');
}
/**
* @param {*} value The value to check.
* @param {function(new: T, ...)} type A user-defined constructor.
* @param {string=} opt_message A message to show when this is hit. * @return {T}

```
 * @template T
 */
/* #export */ function assertInstanceof(value, type, opt_message) {
// We don't use assert immediately here so that we avoid constructing an error //
message if we don't have to.
if (!(value instanceof type)) {
assertNotReached(
opt_message ||
'Value ' + value + ' is not a[n] ' + (type.name || typeof type));
}
return value; }
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.'); </script>
<script>// Copyright (c) 2012 The Chromium Authors. All rights reserved. // Use of this
source code is governed by a BSD-style license that can be // found in the LICENSE
file.
// #import {assertInstanceof} from './assert.m.js'; // #import {dispatchSimpleEvent} from
'./cr.m.js';
/**
 * Alias for document.getElementById. Found elements must be HTMLElements. *
@param {string} id The ID of the element to find.
 * @return {HTMLElement} The found element or null if not found.
 */
/* #export */ function $(id) {
// Disable getElementById restriction here, since we are instructing other // places to re-
use the $() that is defined here.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, HTMLElement) : null;
}
// TODO(devlin): This should return SVGElement, but closure compiler is missing //
those externs.
/**
 * Alias for document.getElementById. Found elements must be SVGElements. *
@param {string} id The ID of the element to find.
 * @return {Element} The found element or null if not found.
 */
/* #export */ function getSVGElement(id) {
// Disable getElementById restriction here, since it is not suitable for SVG // elements.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, Element) : null;
}
/**
 * @return {?Element} The currently focused element (including elements that are *
behind a shadow root), or null if nothing is focused.
 */
```

```
/* #export */ function getDeepActiveElement() {
let a = document.activeElement;
while (a && a.shadowRoot && a.shadowRoot.activeElement) {
a = a.shadowRoot.activeElement; }
return a; }
//
/**
* @param {Node} el A node to search for ancestors with IclassNamel.
* @param {string} className A class to search for.
* @return {Element} A node with class of IclassNamel or null if none is found. */
/* #export */ function findAncestorByClass(el, className) { return /** @type {Element}
*/ (findAncestor(el, function(el) {
return el.classList && el.classList.contains(className); }));
}
/**
```

- 　　　　　 * Return the first ancestor for which the {@code predicate} returns true.

- 　　　　　 * @param {Node} node The node to check.

- 　　　　　 * @param {function(Node):boolean} predicate The function that tests the

- 　　　　　 * nodes.

- 　　　　　 * @param {boolean=} includeShadowHosts

- 　　　　　 * @return {Node} The found ancestor or null if not found. */
  ```
  /* #export */ function findAncestor(node, predicate, includeShadowHosts) { while
  (node !== null) {
  if (predicate(node)) { break;
  }
  node = includeShadowHosts && node instanceof ShadowRoot ? node.host :
  ```

```
}
return node; }
node.parentNode;
/**
* Disables text selection and dragging, with optional callbacks to specify
* overrides.
* @param {function(Event):boolean=} opt_allowSelectStart Unless this function * is
defined and returns true, the onselectionstart event will be
* suppressed.
* @param {function(Event):boolean=} opt_allowDragStart Unless this function * is
defined and returns true, the ondragstart event will be suppressed.
*/
/* #export */ function disableTextSelectAndDrag( opt_allowSelectStart,
opt_allowDragStart) {
```

```
// Disable text selection. document.onselectstart = function(e) {
if (!(opt_allowSelectStart && opt_allowSelectStart.call(this, e))) { e.preventDefault();
} };
// Disable dragging. document.ondragstart = function(e) {
if (!(opt_allowDragStart && opt_allowDragStart.call(this, e))) { e.preventDefault();
} };
}
/**
* Check the directionality of the page.
* @return {boolean} True if Chrome is running an RTL UI. */
/* #export */ function isRTL() {
return document.documentElement.dir === 'rtl';
}
/**
* Get an element that's known to exist by its ID. We use this instead of just * calling
getElementById and not checking the result because this lets us
* satisfy the JSCompiler type system.
* @param {string} id The identifier name.
* @return {!HTMLElement} the Element.
*/
/* #export */ function getRequiredElement(id) { return assertInstanceof(
$(id), HTMLElement, 'Missing required element: ' + id); }
/**
```

- ```
  *  Query an element that's known to exist by a selector. We use this instead of
  ```
- ```
  *  just calling querySelector and not checking the result because this lets us
  ```
- ```
  *  satisfy the JSCompiler type system.
  ```
- ```
  *  @param {string} selectors CSS selectors to query the element.
  ```
- ```
  *  @param {(!Document|!DocumentFragment|!Element)=} opt_context An optional
  ```
- ```
  *  context object for querySelector.
  ```
- ```
  *  @return {!HTMLElement} the Element. */
  /* #export */ function queryRequiredElement(selectors, opt_context) { const element
  = (opt_context || document).querySelector(selectors);
  ```

```
return assertInstanceof(
element, HTMLElement, 'Missing required element: ' + selectors);
}
/**
```

-            * Creates a new URL which is the old URL with a GET param of key=value.

-            * @param {string} url The base URL. There is not sanity checking on the URL so

-            * it must be passed in a proper format.

-            * @param {string} key The key of the param.

-            * @param {string} value The value of the param.

-            * @return {string} The new URL. */

```
/* #export */ function appendParam(url, key, value) {
const param = encodeURIComponent(key) + '=' + encodeURIComponent(value);
if (url.indexOf('?') === -1) { return url + '?' + param;
}
return url + '&' + param; }
/**
* Creates an element of a specified type with a specified class name. * @param
{string} type The node type.
* @param {string} className The class name to use.
* @return {Element} The created element.
*/
/* #export */ function createElementWithClassName(type, className) { const elm =
document.createElement(type);
elm.className = className;
return elm;
}
/**
* transitionend does not always fire (e.g. when animation is aborted
* or when no paint happens during the animation). This function sets up
* a timer and emulate the event if it is not fired when the timer expires.
* @param {!HTMLElement} el The element to watch for transitionend.
* @param {number=} opt_timeOut The maximum wait time in milliseconds for the *
transitionend to happen. If not specified, it is fetched from |el|
* using the transitionDuration style value.
*/
/* #export */ function ensureTransitionEndEvent(el, opt_timeOut) { if (opt_timeOut
=== undefined) {
const style = getComputedStyle(el);
opt_timeOut = parseFloat(style.transitionDuration) * 1000;

// Give an additional 50ms buffer for the animation to complete.
opt_timeOut += 50; }
let fired = false; el.addEventListener('transitionend', function f(e) {
```

```
el.removeEventListener('transitionend', f);
fired = true; });
window.setTimeout(function() { if (!fired) {
cr.dispatchSimpleEvent(el, 'transitionend', true); }
}, opt_timeOut); }
/**
* Alias for document.scrollTop getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The Y document scroll offset.
*/
/* #export */ function scrollTopForDocument(doc) {
return doc.documentElement.scrollTop || doc.body.scrollTop;
}
/**
* Alias for document.scrollTop setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target Y scroll offset.
*/
/* #export */ function setScrollTopForDocument(doc, value)
{ doc.documentElement.scrollTop = doc.body.scrollTop = value;
}
/**
* Alias for document.scrollLeft getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The X document scroll offset.
*/
/* #export */ function scrollLeftForDocument(doc) {
return doc.documentElement.scrollLeft || doc.body.scrollLeft;
}
/**
* Alias for document.scrollLeft setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target X scroll offset.
*/
/* #export */ function setScrollLeftForDocument(doc, value)
{ doc.documentElement.scrollLeft = doc.body.scrollLeft = value;
}
/**
* Replaces '&', '<', '>', '"', and ''' characters with their HTML encoding.
* @param {string} original The original string.
* @return {string} The string with all the characters mentioned above replaced. */
/* #export */ function HTMLEscape(original) { return original.replace(/&/g, '&amp;')
```

```
      .replace(/</g, '&lt;') .replace(/>/g, '&gt;') .replace(/"/g, '&quot;') .replace(/'/g, '&#39;');
}
/**
 * Shortens the provided string (if necessary) to a string of length at most
 * |maxLength|.
 * @param {string} original The original string.
 * @param {number} maxLength The maximum length allowed for the string. * @return
 {string} The original string if its length does not exceed
 * |maxLength|. Otherwise the first |maxLength| - 1 characters with '...'
 * appended.
 */
/* #export */ function elide(original, maxLength) { if (original.length <= maxLength) {
return original; }
return original.substring(0, maxLength - 1) + '\u2026'; }
/**
 * Quote a string so it can be used in a regular expression. * @param {string} str The
 source string.
 * @return {string} The escaped string.
 */
/* #export */ function quoteString(str) {
return str.replace(/([\\\.\+\*\?\[\^\]\$\(\)\{\}\=\!\<\>\|\:])/g, '\\$1'); }
/**
 * Calls |callback| and stops listening the first time any event in |eventNames|
 * is triggered on |target|.
 * @param {!EventTarget} target
 * @param {!Array<string>|string} eventNames Array or space-delimited string of * event
 names to listen to (e.g. 'click mousedown').
 * @param {function(!Event)} callback Called at most once. The
 * optional return value is passed on by the listener.
 */
/* #export */ function listenOnce(target, eventNames, callback) { if (!
Array.isArray(eventNames)) {
eventNames = eventNames.split(/ +/); }
const removeAllAndCallCallback = function(event)
{ eventNames.forEach(function(eventName) {
target.removeEventListener(eventName, removeAllAndCallCallback, false); });
return callback(event); };
eventNames.forEach(function(eventName) { target.addEventListener(eventName,
removeAllAndCallCallback, false);
}); }
/**
 * @param {!Event} e
 * @return {boolean} Whether a modifier key was down when processing |e|. */
/* #export */ function hasKeyModifiers(e) {
return !!(e.altKey || e.ctrlKey || e.metaKey || e.shiftKey);
}
```

```
/**
* @param {!Element} el
* @return {boolean} Whether the element is interactive via text input. */
/* #export */ function isTextInputElement(el) {
return el.tagName === 'INPUT' || el.tagName === 'TEXTAREA';
}
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
</script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
const CAPTIVEPORTAL_CMD_OPEN_LOGIN_PAGE = 'openLoginPage'; </script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
function setupSSLDebuggingInfo() {
if (loadTimeData.getString('type') !== 'SSL') {
return; }
// The titles are not internationalized because this is debugging information
// for bug reports, help center posts, etc.
appendDebuggingField('Subject', loadTimeData.getString('subject'));
appendDebuggingField('Issuer', loadTimeData.getString('issuer'));
appendDebuggingField('Expires on', loadTimeData.getString('expirationDate'));
appendDebuggingField('Current date', loadTimeData.getString('currentDate'));
appendDebuggingField('PEM encoded chain', loadTimeData.getString('pem'),
true);
const ctInfo = loadTimeData.getString('ct');
if (ctInfo) {
appendDebuggingField('Certificate Transparency', ctInfo);
}
$('error-code').addEventListener('click', toggleDebuggingInfo); $('error-
code').setAttribute('role', 'button'); $('error-code').setAttribute('aria-expanded', false);
} </script>
<script>// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h. const SB_BOX_CHECKED =
'boxchecked';
const SB_DISPLAY_CHECK_BOX = 'displaycheckbox';
// This sets up the Extended Safe Browsing Reporting opt-in, either for // reporting
malware or invalid certificate chains. Does nothing if the
// interstitial type is not SAFEBROWSING or SSL or CAPTIVE_PORTAL. function
setupExtendedReportingCheckbox() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' &&
```

```javascript
interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!loadTimeData.getBoolean(SB_DISPLAY_CHECK_BOX)) { return;
}
if ($('privacy-link')) { $('privacy-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY);
return false;
});
$('privacy-link').addEventListener('mousedown', function() {
return false; });
}
$('opt-in-checkbox').checked = loadTimeData.getBoolean(SB_BOX_CHECKED); $
('extended-reporting-opt-in').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-opt-in';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-opt-in'; }
$('extended-reporting-opt-in').classList.add(className);
$('body').classList.add('extended-reporting-has-checkbox');
if ($('whitepaper-link')) { $('whitepaper-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER); });
}
$('opt-in-checkbox').addEventListener('click', function() { sendCommand($('opt-in-
checkbox').checked ?
SecurityInterstitialCommandId.CMD_DO_REPORT :
SecurityInterstitialCommandId.CMD_DONT_REPORT);
}); }
</script>
<script>// Copyright 2020 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h.
const SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE =
'displayEnhancedProtectionMessage';
// This sets up the enhanced protection message. function
setupEnhancedProtectionMessage() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' &&
interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!
loadTimeData.getBoolean(SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE)) {
return; }
if ($('enhanced-protection-link')) { if (mobileNav) {
```

```
// To make sure the touch area of the link is larger than the
// minimum touch area for accessibility, make the whole block tappable. $('enhanced-
protection-message').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
ON_SETTINGS);
return false; });
} else {
$('enhanced-protection-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
ON_SETTINGS);
return false; });
} }
$('enhanced-protection-message').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-enhanced-protection-message';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-enhanced-protection-message'; }
$('enhanced-protection-message').classList.add(className); }
</script>
<script>// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
let mobileNav = false;
/**
* For small screen mobile the navigation buttons are moved * below the advanced text.
*/
function onResize() {
const helpOuterBox = document.querySelector('#details');
const mainContent = document.querySelector('#main-content');
const mediaQuery = '(min-width: 240px) and (max-width: 420px) and ' +
'(min-height: 401px), ' +
'(max-height: 560px) and (min-height: 240px) and ' + '(min-width: 421px)';
const detailsHidden = helpOuterBox.classList.contains(HIDDEN_CLASS); const
runnerContainer = document.querySelector('.runner-container');
// Check for change in nav status.
if (mobileNav !== window.matchMedia(mediaQuery).matches) {
mobileNav = !mobileNav;
// Handle showing the top content / details sections according to state. if (mobileNav) {
mainContent.classList.toggle(HIDDEN_CLASS, !detailsHidden);
helpOuterBox.classList.toggle(HIDDEN_CLASS, detailsHidden); if (runnerContainer) {
runnerContainer.classList.toggle(HIDDEN_CLASS, !detailsHidden); }
} else if (!detailsHidden) {
// Non mobile nav with visible details. mainContent.classList.remove(HIDDEN_CLASS);
helpOuterBox.classList.remove(HIDDEN_CLASS); if (runnerContainer) {
runnerContainer.classList.remove(HIDDEN_CLASS); }
```

```
} }
}
function setupMobileNav() { window.addEventListener('resize', onResize); onResize();
}
document.addEventListener('DOMContentLoaded', setupMobileNav); </script>
```

// This is the shared code for security interstitials. It is used for both SSL // interstitials and Safe Browsing interstitials.
/**

- * @typedef {{

- * dontProceed: function(),

- * proceed: function(),

- * showMoreSection: function(),

- * openHelpCenter: function(),

- * openDiagnostic: function(),

- * reload: function(),

- * openDateSettings: function(),

- * openLogin: function(),

- * doReport: function(),

- * dontReport: function(),

- * openReportingPrivacy: function(),

- * openWhitepaper: function(),

- * reportPhishingError: function(),

- * }} */
  // eslint-disable-next-line no-var var certificateErrorPageController;
  // Should match security_interstitials::SecurityInterstitialCommand /** @enum {number} */
  const SecurityInterstitialCommandId = {

CMD_DONT_PROCEED: 0,

```javascript
CMD_PROCEED: 1,
// Ways for user to get more information CMD_SHOW_MORE_SECTION: 2,
CMD_OPEN_HELP_CENTER: 3, CMD_OPEN_DIAGNOSTIC: 4,
// Primary button actions
CMD_RELOAD: 5,
CMD_OPEN_DATE_SETTINGS: 6,
CMD_OPEN_LOGIN: 7,
// Safe Browsing Extended Reporting
CMD_DO_REPORT: 8,
CMD_DONT_REPORT: 9, CMD_OPEN_REPORTING_PRIVACY: 10,
CMD_OPEN_WHITEPAPER: 11,
// Report a phishing error. CMD_REPORT_PHISHING_ERROR: 12,
// Open enhanced protection settings.
CMD_OPEN_ENHANCED_PROTECTION_SETTINGS: 13,
};
const HIDDEN_CLASS = 'hidden';
/**
 * A convenience method for sending commands to the parent page.
 * @param {SecurityInterstitialCommandId} cmd The command to send. */
function sendCommand(cmd) {
if (window.certificateErrorPageController) {
switch (cmd) {
case SecurityInterstitialCommandId.CMD_DONT_PROCEED:
certificateErrorPageController.dontProceed();
break;
case SecurityInterstitialCommandId.CMD_PROCEED:
certificateErrorPageController.proceed();
break;
case SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION:
certificateErrorPageController.showMoreSection();
break;
case SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER:
certificateErrorPageController.openHelpCenter();
break;
case SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC:
certificateErrorPageController.openDiagnostic();
break;
case SecurityInterstitialCommandId.CMD_RELOAD:
certificateErrorPageController.reload();
break;
case SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS:
certificateErrorPageController.openDateSettings();
break;
case SecurityInterstitialCommandId.CMD_OPEN_LOGIN:
certificateErrorPageController.openLogin();
break;
```

```
case SecurityInterstitialCommandId.CMD_DO_REPORT:
certificateErrorPageController.doReport();
break;
case SecurityInterstitialCommandId.CMD_DONT_REPORT:
certificateErrorPageController.dontReport();
break;
case SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY:
certificateErrorPageController.openReportingPrivacy();
break;
case SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER:
certificateErrorPageController.openWhitepaper();
break;
case SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR:
certificateErrorPageController.reportPhishingError();
break; case
SecurityInterstitialCommandId.CMD_OPEN_ENHANCED_PROTECTION_SETTINGS:
certificateErrorPageController.openEnhancedProtectionSettings();
break;
}
return; }
// window.domAutomationController.send(cmd); //
//
}
/**
* Call this to stop clicks on <a href="#"> links from scrolling to the top of * the page (and
possibly showing a # in the link).
*/
function preventDefaultOnPoundLinkClicks() { document.addEventListener('click',
function(e) {
const anchor = findAncestor(/** @type {Node} */ (e.target), function(el) { return
el.tagName === 'A';
});
// Use getAttribute() to prevent URL normalization. if (anchor &&
anchor.getAttribute('href') === '#') {
e.preventDefault(); }
}); }
//
// </script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// This is the shared code for the new (Chrome 37) security interstitials. It is // used for
both SSL interstitials and Safe Browsing interstitials.
let expandedDetails = false; let keyPressState = 0;
/**
* This allows errors to be skippped by typing a secret phrase into the page. * @param
```

{string} e The key that was just pressed.
*/
function handleKeypress(e) {
// HTTPS errors are serious and should not be ignored. For testing purposes, // other approaches are both safer and have fewer side-effects.
// See https://goo.gl/ZcZixP for more details.
const BYPASS_SEQUENCE = window.atob('dGhpc2lzdW5zYWZl');
if (BYPASS_SEQUENCE.charCodeAt(keyPressState) === e.keyCode) {
keyPressState++;
if (keyPressState === BYPASS_SEQUENCE.length) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED);
keyPressState = 0; }
} else { keyPressState = 0;
} }
/**
* This appends a piece of debugging information to the end of the warning. * When complete, the caller must also make the debugging div
* (error-debugging-info) visible.
* @param {string} title The name of this debugging field.
* @param {string} value The value of the debugging field.
* @param {boolean=} fixedWidth If true, the value field is displayed fixed
* width.
*/
function appendDebuggingField(title, value, fixedWidth) {
// The values input here are not trusted. Never use innerHTML on these // values!
const spanTitle = document.createElement('span'); spanTitle.classList.add('debugging-title');
spanTitle.innerText = title + ': ';
const spanValue = document.createElement('span');
spanValue.classList.add('debugging-content');
if (fixedWidth) {
spanValue.classList.add('debugging-content-fixed-width'); }
spanValue.innerText = value;
const pElem = document.createElement('p'); pElem.classList.add('debugging-content');
pElem.appendChild(spanTitle); pElem.appendChild(spanValue); $('error-debugging-info').appendChild(pElem);
}
function toggleDebuggingInfo() {
const hiddenDebug = $('error-debugging-info').classList.toggle(HIDDEN_CLASS); $('error-code').setAttribute('aria-expanded', !hiddenDebug);
}
function setupEvents() {
const overridable = loadTimeData.getBoolean('overridable'); const interstitialType = loadTimeData.getString('type');
const ssl = interstitialType === 'SSL';
const captivePortal = interstitialType === 'CAPTIVE_PORTAL'; const badClock = ssl &&

```javascript
loadTimeData.getBoolean('bad_clock'); const lookalike = interstitialType ===
'LOOKALIKE';
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing'); const
originPolicy = interstitialType === 'ORIGIN_POLICY';
const blockedInterception = interstitialType === 'BLOCKED_INTERCEPTION'; const
legacyTls = interstitialType == 'LEGACY_TLS';
const insecureForm = interstitialType == 'INSECURE_FORM';
const httpsOnly = interstitialType == 'HTTPS_ONLY';
const hidePrimaryButton = loadTimeData.getBoolean('hide_primary_button'); const
showRecurrentErrorParagraph = loadTimeData.getBoolean(
'show_recurrent_error_paragraph');
if (ssl || originPolicy || blockedInterception || legacyTls) { $
('body').classList.add(badClock ? 'bad-clock' : 'ssl');
if (loadTimeData.valueExists('errorCode')) {
$('error-code').textContent = loadTimeData.getString('errorCode');
$('error-code').classList.remove(HIDDEN_CLASS); }
```

- ```javascript
  } else if (captivePortal) { $('body').classList.add('captive-portal');
  ```

- ```javascript
  } else if (billing) { $('body').classList.add('safe-browsing-billing');
  ```

- ```javascript
  } else if (lookalike) { $('body').classList.add('lookalike-url');
  ```

- ```javascript
  } else if (insecureForm) { $('body').classList.add('insecure-form');
  ```

- ```javascript
  } else if (httpsOnly) { $('body').classList.add('https-only');
  ```

- ```javascript
  } else {
  ```
  ```javascript
  $('body').classList.add('safe-browsing');
  // Override the default theme color. document.querySelector('meta[name=theme-
  color]').setAttribute('content',
  'rgb(217, 48, 37)'); }
  $('icon').classList.add('icon');
  if (legacyTls) { $('icon').classList.add('legacy-tls');
  }
  if (hidePrimaryButton) { $('primary-button').classList.add(HIDDEN_CLASS);
  } else {
  $('primary-button').addEventListener('click', function() {
  switch (interstitialType) { case 'CAPTIVE_PORTAL':
  sendCommand(SecurityInterstitialCommandId.CMD_OPEN_LOGIN); break;
  case 'SSL':
  case 'LEGACY_TLS':
  if (badClock) {
  sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS); }
  else if (overridable) {
  sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); } else {
  ```

```
sendCommand(SecurityInterstitialCommandId.CMD_RELOAD); }
break;
case 'SAFEBROWSING': case 'ORIGIN_POLICY':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
break;
case 'HTTPS_ONLY': case 'INSECURE_FORM': case 'LOOKALIKE':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); break;
default:
throw 'Invalid interstitial type';
} });
}
if (lookalike || insecureForm || httpsOnly) {
const proceedButton = 'proceed-button'; $
(proceedButton).classList.remove(HIDDEN_CLASS); $(proceedButton).textContent =
loadTimeData.getString('proceedButtonText'); $
(proceedButton).addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
}
if (lookalike) {
// Lookalike interstitials with a suggested URL have a link in the title:
// "Did you mean <link>example.com</link>?". Handle those clicks. Lookalike //
interstitails without a suggested URL don't have this link.
const dontProceedLink = 'dont-proceed-link';
if ($(dontProceedLink)) {
$(dontProceedLink).addEventListener('click', function(event)
{ sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
}); }
}
if (overridable) {
const overrideElement = billing ? 'proceed-button' : 'proceed-link'; // Captive portal page
isn't overridable. $(overrideElement).addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
if (ssl) { $(overrideElement).classList.add('small-link');
} else if (billing) { $(overrideElement).classList.remove(HIDDEN_CLASS); $
(overrideElement).textContent =
loadTimeData.getString('proceedButtonText'); }
} else if (!ssl) { $('final-paragraph').classList.add(HIDDEN_CLASS);
}
if (!ssl || !showRecurrentErrorParagraph) { $('recurrent-error-
message').classList.add(HIDDEN_CLASS);
} else { $('body').classList.add('showing-recurrent-error-message');
}
if ($('diagnostic-link')) { $('diagnostic-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC); });
}
```

```
if ($('learn-more-link')) { $('learn-more-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER); });
}
if (captivePortal || billing || lookalike || insecureForm || httpsOnly) { // Captive portal,
billing, lookalike pages, insecure form, and
// HTTPS only mode interstitials don't have details buttons. $('details-
button').classList.add('hidden');
} else { $('details-button')
.setAttribute(
'aria-expanded', !$('details').classList.contains(HIDDEN_CLASS));
$('details-button').addEventListener('click', function(event) {
const hiddenDetails = $('details').classList.toggle(HIDDEN_CLASS); $('details-
button').setAttribute('aria-expanded', !hiddenDetails);
if (mobileNav) {
// Details appear over the main content on small screens. $('main-
content').classList.toggle(HIDDEN_CLASS, !hiddenDetails);
} else { $('main-content').classList.remove(HIDDEN_CLASS);
}
$('details-button').innerText = hiddenDetails ? loadTimeData.getString('openDetails') :
loadTimeData.getString('closeDetails');
if (!expandedDetails) {
// Record a histogram entry only the first time that details is opened.
sendCommand(SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION);
expandedDetails = true;
} });
}
if ($('report-error-link')) { $('report-error-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR); })
;
}
if (lookalike) { console.log(
'Chrome has determined that ' + loadTimeData.getString('lookalikeRequestHostname')
+ ' could be fake or fraudulent.\n\n' +
'If you believe this is shown in error please visit ' + 'https://g.co/chrome/lookalike-
warnings');
}
preventDefaultOnPoundLinkClicks(); setupExtendedReportingCheckbox();
setupEnhancedProtectionMessage(); setupSSLDebuggingInfo();
document.addEventListener('keypress', handleKeypress);
}
document.addEventListener('DOMContentLoaded', setupEvents); </script>
</head>
<body id="body" class="ssl">
<div class="interstitial-wrapper">
<div id="main-content" class="hidden">
<div class="icon" id="icon"></div> <div id="main-message">
```

```html
<h1>Your connection is not private</h1>
<p>Attackers might be trying to steal your information from
<strong>accounts.shopify.com</strong> (for example, passwords, messages, or credit
cards). <a href="#" id="learn-more-link">Learn more</a></p>
<div id="debugging">
<div id="error-code" class="error-code" role="button" aria-
expanded="false">net::ERR_CERT_DATE_INVALID</div>
<div id="error-debugging-info" class="hidden"><p class="debugging-
content"><span class="debugging-title">Subject: </span><span class="debugging-
content">accounts.shopify.com</span></p><p class="debugging-content"><span
class="debugging-title">Issuer: </span><span class="debugging-content">R3</span></
p><p class="debugging-content"><span class="debugging-title">Expires on: </
span><span class="debugging-content">Nov 13, 2021</span></p><p
class="debugging-content"><span class="debugging-title">Current date: </span><span
class="debugging-content">Oct 5, 2021</span></p><p class="debugging-
content"><span class="debugging-title">PEM encoded chain: </span><span
class="debugging-content debugging-content-fixed-width">-----BEGIN
CERTIFICATE-----<br>MIIF2DCCBMCgAwIBAgISA3lP
+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA<br>MDIxCzAJBgNVBAYTAlV
TMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQYDVQQD<br>EwJSMzAeFw
0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNVBAMT<br>FGFj
Y291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB<br>CgK
CAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw<br>gkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw<br>5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH<br>w2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/
vtIs<br>ErLP9noTy1wdKzXJYbu4lT6nVOSs
+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX<br>MwPm/Opq3qk3BnI5uZgyV
+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/
BAQD<br>AgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBg
NVHRMBAf8EAjAA<br>MB0GA1UdDgQWBBR/TRz5xF8Ma/
wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX<br>t1hWy65QCUDmH6+dixTCxjBVB
ggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGFWh0<br>dHA6Ly9yMy5vLmxlbmN
yLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkubGVu<br>Y3Iub3JnLzCByQYDV
R0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91<br>ZC5jb22CFGFjY291bnRzLnNob3B
pZnkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w<br>aWZ5Y2xvdWQuY29tgiZpZGVudGl0
eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk<br>LmNvbYIiaWRlbnRpdHktbWVyY2h
hbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp<br>dHkuc2hvcGlmeWNsb3VkLmNv
bTBMBgNVHSAERTBDMAgGBmeBDAECATA3BgsrBgEE<br>AYLfEwEBATAoMCYGC
CsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0Lm9y<br>ZzCCAQMGCisGA
QQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj+KMDa5oK+<br>2MsxtT/
TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx<br>HBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14ITkrE<br>AHYAfT7y+I//
```

iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA
RzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh
+q
AgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQ
BcKanD
DWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWIfOU
LxGoEd/k
SA0t9X94AFhu+FhBQnrssUH/D
+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
dWfgQcEhx3HInYVT
+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h
Z8Y/kug
+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5
MVXdjQTeKuziIyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr
/+lfVfwGqqjAuskK
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/
bWLiWnX1owDQYJKoZIhvcNAQELBQAw
TzELMAkGA1UEBhMCVVMxKTAnBgN
VBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQD
EwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDAwMDAw
WhcNMjUwOTE1MTYw
MDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChMNTGV0J3Mg
RW5jcnl
wdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK
AoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
R5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCU
JCWH+jdx
sxPnHKzhm+/
b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
NHz6a4uPVy
mZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
Z3Vms/
EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG
/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
AYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
Af8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8
GA1UdIwQYMBaA
FHm0WeZ7tuXkAXOACIjlGlj26ZtuMDIGCCsGAQUFBwEBBCY
wJDAiBggrBgEFBQcw
AoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EI
DAeMBygGqAYhhZodHRw
Oi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCA
YGZ4EMAQIBMA0GCysGAQQB
gt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQ
CFyk5HPqP3hUSFvNVneLKYY611TR6W
PTNlclQtgaDqw+34IL9fzLdwALduO/
ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
ikfmZW4/RvTZ8M6UK
+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
CkcheAmCJ8MqyJu
5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh
+Blm
lJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4
avAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2
yJMC6alLbfODALZvYH7n7do1AZls4I9d1P4jnkDrQox
B3UqQ9hVl3LEKQ73xF1O
yK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
hCExroL1+7mryIkXPeF
M5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
HlUjr8gRsI3qfJOQFy/
9rKlJR0Y/8Omwt/
8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
MldITTKB3zhThV1+XWYp6rjd5JW1zbVW
EkLNxE7GJThEUG3szgBVGP7pSWTUTsqX
nLRbwHOoq7hHwg==
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFYDCCBEigAwIBAgIQQAF3ITfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFAD
A/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMD
E4MTQwM1ow
TzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY
3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwg
gIiMA0GCSqGSIb3DQEB
AQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHI
e+3LffOJCMbjzmV6B493XC
ov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL
wYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+IAOf00eXfJlII1PoOK5PCm
+D
LtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P
+1dy70sntK
4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5
bHbvO5BieebbpJovJsXQEOEO3tkQjhb
7t/eo98flAgeYjzYIlefiN5YNNnWe+w5y
sR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ
Xmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4
FQsDj43QLwWy
PntKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
SLeCO5imfW
CKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/
H2ql
PRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/
7xk9TND
TwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
SwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUF
BzAChi9odHRwOi8vYXBwcy5pZGVudHJ1
c3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgz
LnA3YzAfBgNVHSMEGDAWgBTEp7Gkeyxx
+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
ATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3Qte
DEubGV0c2VuY3J5cHQu
b3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9
jcmwuaWRlbnRydXN0LmNvbS9E
U1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OB
BYEFHm0WeZ7tuXkAXOACIjlGlj26Ztu
MA0GCSqGSIb3DQEBCwUAA4IBAQAKcw
Bslm7/DILQrt2M51oGrS+o44+/
yQoDFVDC
5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7
G75K/
QosMrAdSW
9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
WCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6U
SbW1FzXSLr9O
he8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE
+FiYruq9RUWhiF1myv4Q6W
+CyBFC
Dfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5

END CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE
0MDExNVow
PzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ
28uMRcwFQYDVQQD
Ew5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNA
QEBBQADggEPADCCAQoCggEB
AN
+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O
rz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
OLl5CjH9
UL2AZd

+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
xiqKqy69cK3FCx
olkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/
yUFw
7BZy1SbsOFU5Q9D8/
RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
aeQQmxkqtilX4+U9m
5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV
HQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG
SIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmgISBd49IZRNI+DT69
ikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX
+5v3gTt23ADq1cEmv8uXr
AvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbu
mR9YbK
+rlmM6pZW87ipxZz
R8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5
JDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo
Ob8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ<b
r>-----END CERTIFICATE-----
</p><p class="debugging-content">Certificate Transparency: <span class="debugging-
content">

SCT Cloudflare 'Nimbus2021' Log (Embedded in certificate,
Verified)

SCT Google 'Xenon2021' log (Embedded in certificate, Verified)</
span></p></div>
</div>
</div>
<div id="recurrent-error-message" class="hidden">
Warnings may be common while websites update their security. This should improve
soon.
</div>
<div id="extended-reporting-opt-in" class="hidden">
<label>
<div class="checkboxes">
<input type="checkbox" id="opt-in-checkbox">

</div>
Help improve security on the web for everyone by
sending URLs of some pages you visit, limited system
information, and some page content to Google. <a id="privacy-link"
href="#">Privacy policy
</label>
</div>
<div id="enhanced-protection-message" class="hidden">
<label>
<div class="icon"></div>
<div>To get Chrome's highest level of security, <a href="#" id="enhanced-
protection-link">turn on enhanced protection</div> </label>
</div>
</div>
<div class="nav-wrapper">
<button id="primary-button">Reload</button>

<button id="proceed-button" class="secondary-button small-link hidden"></button>
<button id="details-button" class="secondary-button small-link" aria-expanded="true">Hide advanced</button> </div>
<div id="details" class="">
<p>accounts.shopify.com normally uses encryption to protect your information. When Chrome tried to connect to accounts.shopify.com this time, the website sent back unusual and incorrect credentials. This may happen when an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.</p>
<p id="final-paragraph">You cannot visit accounts.shopify.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.</p>
</div> </div>
<style>/* Copyright 2014 The Chromium Authors. All rights reserved.

- * Use of this source code is governed by a BSD-style license that can be

- * found in the LICENSE file. */

/* This file is dynamically processed by a C++ data source handler to fill in
- * some per-platform/locale styles that dramatically alter the page. This is

- * done to reduce flicker, as JS may not run before the page is rendered.
  *

- * There are two ways to include this stylesheet:

- * 1. via its chrome://resources/ URL in HTML, i.e.: *

- * <link rel="stylesheet" href="chrome://resources/css/text_defaults.css">
  *

- * 2. via the webui::AppendWebUICSSTextDefaults() method to directly append it

- * to an HTML string.

- * Otherwise its placeholders won't be expanded. */
    body {
    font-family: system-ui, sans-serif; font-size: 75%;
    }
    button {
    font-family: system-ui, sans-serif;
    }
    </style><script>// Copyright (c) 2012 The Chromium Authors. All rights reserved. // Use of this source code is governed by a BSD-style license that can be

// found in the LICENSE file.
/**

- * @fileoverview This file defines a singleton which provides access to all data

- * that is available as soon as the page's resources are loaded (before DOM

- * content has finished loading). This data includes both localized strings and

- * any data that is important to have ready from a very early stage (e.g. things

- * that must be displayed right away). *

- * Note that loadTimeData is not guaranteed to be consistent between page

- * refreshes (https://crbug.com/740629) and should not contain values that might

- * change if the page is re-opened later. */
/** @type {!LoadTimeData} */
// eslint-disable-next-line no-var var loadTimeData;
class LoadTimeData { constructor() {
/** @type {?Object} */
this.data_ = null; }
/**
  - * Sets the backing object. *

  - * Note that there is no getter for |data_| to discourage abuse of the form: *

  - * var value = loadTimeData.data()['key']; *

  - * @param {Object} value The de-serialized page data. */
  set data(value) {
  expect(!this.data_, 'Re-setting data.'); this.data_ = value;
  }
  /**
  * @param {string} id An ID of a value that might exist.
  * @return {boolean} True if |id| is a key in the dictionary. */
  valueExists(id) {
  return id in this.data_;

```
      }

/**
 * Fetches a value, expecting that it exists.
 * @param {string} id The key that identifies the desired value. * @return {*} The
corresponding value.
 */
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**
 * As above, but also makes sure that the value is a string.
 * @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
 */
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
 *                      *  Returns a formatted localized string where $1 to $9 are replaced by the

 *                      *  second to the tenth argument.

 *                      *  @param {string} id The ID of the string we want.

 *                      *  @param {...(string|number)} var_args The extra values to include in
   the

 *                      *  formatted output.

 *                      *  @return {string} The formatted string. */
   getStringF(id, var_args) {
   const value = this.getString(id); if (!value) {
   return ''; }
   const args = Array.prototype.slice.call(arguments); args[0] = value;
   return this.substituteString.apply(this, args);
   }
   /**
   * Returns a formatted localized string where $1 to $9 are replaced by the
   * second to the tenth argument. Any standalone $ signs must be escaped as * $$.

 * @param {string} label The label to substitute through.
 * This is not an resource ID.
```

```
* @param {...(string|number)} var_args The extra values to include in the * formatted
output.
* @return {string} The formatted string.
*/
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
} /**
* * * * * * * * * * */
```

Returns a formatted string where $1 to $9 are replaced by the second to tenth
argument, split apart into a list of pieces describing how the substitution was performed.
Any standalone $ signs must be escaped as $$. @param {string} label A localized string
to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the
formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted
string pieces.

```
getSubstitutedStringPieces(label, var_args) {
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9
pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
[]).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
* As above, but also makes sure that the value is a boolean.
* @param {string} id The key that identifies the desired boolean. * @return {boolean}
The corresponding boolean value.
*/
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type
{boolean} */ (value);
}
/**
* As above, but also makes sure that the value is an integer.
* @param {string} id The key that identifies the desired number. * @return {number} The
```

```
corresponding number value.
*/
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
{number} */ (value);
}
/**
* Override values in loadTimeData with the values found in |replacements|. * @param
{Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) { expect(
typeof replacements === 'object',
'Replacements must be a dictionary object.'); for (const key in replacements) {
this.data_[key] = replacements[key]; }
}
/**
* Reset loadTimeData's data. Should only be used in tests.
* @param {?Object} newData The data to restore to, when null restores to * unset state.
*/
resetForTesting(newData = null) { this.data_ = newData;
}
/**
* @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
* Checks condition, throws error message if expectation fails.
* @param {*} condition The condition to check for truthiness.
* @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
} }
/**
* Checks that the given value has the given type.
* @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
* @param {string} type The type we expect |value| to be.
*/
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
```

local and not all files have been updated to import the exported // IloadTimeDatal explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');</script><script>loadTimeData.data = {"bad_clock":false,"closeDetails":"Hide advanced","ct":"\n\nSCT Cloudflare 'Nimbus2021' Log (Embedded in certificate, Verified)\n\nSCT Google 'Xenon2021' log (Embedded in certificate, Verified)","currentDate":"Oct 5, 2021","displayEnhancedProtectionMessage":false,"displaycheckbox":false,"enhancedProtectionMessage":"To get Chrome's highest level of security, \u003Ca href=\"#\" id=\"enhanced-protection-link\">turn on enhanced protection\u003C/a>","errorCode":"net::ERR_CERT_DATE_INVALID","errorType":1,"expirationDate":"Nov 13, 2021","explanationParagraph":"accounts.shopify.com normally uses encryption to protect your information. When Chrome tried to connect to accounts.shopify.com this time, the website sent back unusual and incorrect credentials. This may happen when an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.","finalParagraph":"You cannot visit accounts.shopify.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.","fontfamily":"system-ui, sans-serif","fontsize":"75%","heading":"Your connection is not private","hide_primary_button":false,"issuer":"R3","language":"en","openDetails":"Advanced","optInLink":"Help improve security on the web for everyone by sending \u003Ca href=\"#\" id=\"whitepaper-link\">URLs of some pages you visit, limited system information, and some page content\u003C/a> to Google. \u003Ca id=\"privacy-link\" href=\"#\">Privacy policy\u003C/a>","overridable":false,"pem":"-----BEGIN CERTIFICATE-----\nMIIF2DCCBMCgAwIBAgISA3lP+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA\nMDIxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQYDVQQD\nEwJSMzAeFw0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNVBAMT\nFGFjY291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB\nCgKCAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/N7mdStXBNjXN9WAEbLSTGrw\ngkyShj1dbDXVJudJ7YdQXJrl/KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw\n5Zgi0Sojj6d87SEqpS2i0Vz6V+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH\nnw2S2OFtEJxH3SSoKfsA/ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/vtIs\nErLP9noTy1wdKzXJYbu4lT6nVOSs+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX\nMwPm/Opq3qk3BnI5uZgyV+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/BAQD\nAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBgNVHRMBAf8EAjAA\nMB0GA1UdDgQWBBR/TRz5xF8Ma/wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX\nt1hWy65QCUDmH6+dixTCxjBVBggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGF

Wh0\ndHA6Ly9yMy5vLmxlbmNyLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkub
GVu\nY3Iub3JnLzCByQYDVR0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91\nZC5jb22CFGFjY291bnRzLnNob3BpZ
nkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w
\naWZ5Y2xvdWQuY29tgiZpZGVudGl0eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk
\nLmNvbYIiaWRlbnRpdHktbWVyY2hhbnQuc2hvcGlmeWNsb3VkLmNvYIZaWRlbnRp
\ndHkuc2hvcGlmeWNsb3VkLmNvbTBMBgNVHSAERTBDMAgGBmeBDAECATA3Bgsr
BgEE
\nAYLfEwEBATAoMCYGCCsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0L
m9y\nZzCCAQMGCisGAQQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj
+KMDa5oK+\n2MsxtT/TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx\nHBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14lTkrE\nAHYAfT7y+I//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA\nRzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh+q
\nAgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQBcKa
nD\nDWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWlfOULxGoEd/
k\nSA0t9X94AFhu+FhBQnrssUH/D+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
\ndWfgQcEhx3HInYVT+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h\nZ8Y/
kug+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5\nMVXdjQTeKuziIyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr\n/+lfVfwGqqjAuskK\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/bWLiWnX1owDQYJKoZIhvcNAQELBQAw
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDA
wMDAw
\nWhcNMjUwOTE1MTYwMDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChM
NTGV0J3Mg
\nRW5jcnlwdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEK\nAoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
\nR5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCUJCWH+jdx
\nsxPnHKzhm+/b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
\nNHz6a4uPVymZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
\nZ3Vms/EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG\n/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
\nAYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
\nAf8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8GA1UdIwQY
MBaA
\nFHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCYwJDAiBggrBgEFBQc
w
\nAoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EIDAeMBygGqAYhhZodH
Rw
\nOi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCAYGZ4EMAQIBMA0GCysGAQ

QB

\ngt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQCFyk5HPqP3hUSFvNVneLKYY611T
R6W\nPTNlclQtgaDqw+34IL9fzLdwALduO/ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
\nikfmZW4/RvTZ8M6UK+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
\nCkcheAmCJ8MqyJu5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh+BIm
\nlJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4\navAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2\nyJMC6alLbBfODALZvYH7n7do1AZls4l9d1P4jnkDrQoxB3
UqQ9hVl3LEKQ73xF1O\nyK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
\nhCExroL1+7mryIkXPeFM5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/IxPFin+
\nHIUjr8gRsl3qfJOQFy/9rKIJR0Y/8Omwt/8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
\nMldlTTKB3zhThV1+XWYp6rjd5JW1zbVWEkLNxE7GJThEUG3szgBVGP7pSWTUTsq
X\nnLRbwHOoq7hHwg==\n-----END CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFYDCCBEigAwIBAgIQQAF3lTfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMDE4MTQw
M1ow
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwggIiMA0GCSqGSIb3D
QEB\nAQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHIe
+3LffOJCMbjzmV6B493XC\nov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL\nwYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlll1PoOK5PCm+D
\nLtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P+1dy70sntK
\n4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5\nbHbvO5BieebbpJovJsXQEOEO3tkQjhb7t/
eo98flAgeYjzYIlefiN5YNNnWe+w5y\nsR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ\nXmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4\nFQsDj43QLwWyPn
tKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
\nSLeCO5imfWCKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/H2ql
\nPRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/7xk9TND
\nTwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
\nSwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUFBzAChi9odHRwOi8vYXBwcy5pZGVu
dHJ1\nc3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgzLnA3YzAfBgNVHSMEGDAWgBTEp7G
keyxx\n+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
\nATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3QteDEubGV0c2VuY3J5cHQ
u
\nb3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9jcmwuaWRlbnRydXN0LmNv
bS9E
\nU1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OBBYEFHm0WeZ7tuXkAXOACIjIGIj26
Ztu\nMA0GCSqGSIb3DQEBCwUAA4IBAQAKcwBslm7/DILQrt2M51oGrS+o44+/

yQoDFVDC
\n5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7G75K/
QosMrAdSW\n9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
\nWCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6USbW1FzXSLr9O
\nhe8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE+FiYruq9RUWhiF1myv4Q6W
+CyBFC\nDfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----\nMIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE0MDExN
Vow
\nPzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ28uMRcwFQYD
VQQD
\nEw5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQ
oCggEB\nAN+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O\nrz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
\nOLl5CjH9UL2AZd+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
\nxiqKqy69cK3FCxolkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/yUFw
\n7BZy1SbsOFU5Q9D8/RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
\naeQQmxkqtilX4+U9m5/wAI0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV\nHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG\nSIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNl+DT69\nikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX+5v3gTt23ADq1cEmv8uXr
\nAvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbumR9YbK+rlmM6pZW87ipxZz
\nR8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5\nJDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo\nOb8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ\n-----
END CERTIFICATE-----\n","primaryButtonText":"Reload","primaryParagraph":"Attackers
might be trying to steal your information from \u003Cstrong>accounts.shopify.com
\u003C/strong> (for example, passwords, messages, or credit cards). \u003Ca href=\"#
\" id=\"learn-more-link\">Learn more\u003C/a>","recurrentErrorParagraph":"Warnings
may be common while websites update their security. This should improve
soon.","show_recurrent_error_paragraph":false,"subject":"accounts.shopify.com","tabTitl
e":"Privacy error","textdirection":"ltr","type":"SSL"};</script></body></html>
————————————————————————
<head>
<meta charset="utf-8">
<meta name="color-scheme" content="light dark">
<meta name="theme-color" content="#fff">
<meta name="viewport" content="initial-scale=1, minimum-scale=1, width=device-width">
<meta http-equiv="Content-Security-Policy" content="require-trusted-types-for 'script';

trusted-types;">
<title>Privacy error</title>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the LICENSE file. */
a{
color: var(--link-color);
}
body {
--background-color: #fff;
--error-code-color: var(--google-gray-700);
--google-blue-100: rgb(210, 227, 252);
--google-blue-300: rgb(138, 180, 248);
--google-blue-600: rgb(26, 115, 232);
--google-blue-700: rgb(25, 103, 210);
--google-gray-100: rgb(241, 243, 244);
--google-gray-300: rgb(218, 220, 224);
--google-gray-500: rgb(154, 160, 166);
--google-gray-50: rgb(248, 249, 250);
--google-gray-600: rgb(128, 134, 139);
--google-gray-700: rgb(95, 99, 104);
--google-gray-800: rgb(60, 64, 67);
--google-gray-900: rgb(32, 33, 36);
--heading-color: var(--google-gray-900);
--link-color: rgb(88, 88, 88); --popup-container-background-color: rgba(0,0,0,.65); --primary-button-fill-color-active: var(--google-blue-700); --primary-button-fill-color: var(--google-blue-600); --primary-button-text-color: #fff;
--quiet-background-color: rgb(247, 247, 247); --secondary-button-border-color: var(--google-gray-500); --secondary-button-fill-color: #fff; --secondary-button-hover-border-color: var(--google-gray-600); --secondary-button-hover-fill-color: var(--google-gray-50); --secondary-button-text-color: var(--google-gray-700); --small-link-color: var(--google-gray-700);
--text-color: var(--google-gray-700);
background: var(--background-color);
color: var(--text-color);
word-wrap: break-word;
}
.nav-wrapper .secondary-button {
background: var(--secondary-button-fill-color);
border: 1px solid var(--secondary-button-border-color); color: var(--secondary-button-text-color);
float: none;
margin: 0;
padding: 8px 16px;
}
.hidden { display: none;

```css
}
html {
-webkit-text-size-adjust: 100%; font-size: 125%;
}
.icon {
background-repeat: no-repeat; background-size: 100%;
}
@media (prefers-color-scheme: dark) { body {
--background-color: var(--google-gray-900); --error-code-color: var(--google-gray-500);
--heading-color: var(--google-gray-500);
--link-color: var(--google-blue-300); --primary-button-fill-color-active: rgb(129, 162, 208);
--primary-button-fill-color: var(--google-blue-300); --primary-button-text-color: var(--
google-gray-900); --quiet-background-color: var(--background-color); --secondary-
button-border-color: var(--google-gray-700); --secondary-button-fill-color: var(--google-
gray-900); --secondary-button-hover-fill-color: rgb(48, 51, 57); --secondary-button-text-
color: var(--google-blue-300); --small-link-color: var(--google-blue-300);
--text-color: var(--google-gray-500); }
} </style>
<style>/* Copyright 2014 The Chromium Authors. All rights reserved. Use of this source
code is governed by a BSD-style license that can be found in the LICENSE file. */
button {
border: 0;
border-radius: 4px;
box-sizing: border-box;
color: var(--primary-button-text-color); cursor: pointer;
float: right;
font-size: .875em;
margin: 0;
padding: 8px 16px;
transition: box-shadow 150ms cubic-bezier(0.4, 0, 0.2, 1); user-select: none;
}
[dir='rtl'] button { float: left;
}
.bad-clock button, .captive-portal button, .https-only button, .insecure-form
button, .lookalike-url button, .main-frame-blocked button, .neterror button,
.pdf button,
.ssl button, .safe-browsing-billing button {
background: var(--primary-button-fill-color); }
button:active {
background: var(--primary-button-fill-color-active); outline: 0;
}
#debugging { display: inline; overflow: auto;
}
.debugging-content { line-height: 1em; margin-bottom: 0; margin-top: 1em;
}
.debugging-content-fixed-width { display: block;
```

```css
font-family: monospace; font-size: 1.2em;
margin-top: 0.5em; }
.debugging-title { font-weight: bold;
}
#details {
margin: 0 0 50px;
}
#details p:not(:first-of-type) { margin-top: 20px;
}
.secondary-button:active {
border-color: white;
box-shadow: 0 1px 2px 0 rgba(60, 64, 67, .3),
0 2px 6px 2px rgba(60, 64, 67, .15); }
.secondary-button:hover {
background: var(--secondary-button-hover-fill-color); border-color: var(--secondary-
button-hover-border-color); text-decoration: none;
}
.error-code {
color: var(--error-code-color); font-size: .8em;
margin-top: 12px; text-transform: uppercase;
}
#error-debugging-info { font-size: 0.8em;
}
h1 {
color: var(--heading-color); font-size: 1.6em; font-weight: normal; line-height: 1.25em;
margin-bottom: 16px;
}
h2 {
font-size: 1.2em; font-weight: normal;
}
.icon {
height: 72px; margin: 0 0 40px; width: 72px;
}
input[type=checkbox] { opacity: 0;
}
input[type=checkbox]:focus ~ .checkbox::after { outline: -webkit-focus-ring-color auto
5px;
}
.interstitial-wrapper { box-sizing: border-box; font-size: 1em; line-height: 1.6em; margin:
14vh auto 0; max-width: 600px; width: 100%;
}
#main-message > p { display: inline;
}
#extended-reporting-opt-in { font-size: .875em; margin-top: 32px;
}
#extended-reporting-opt-in label { display: grid; grid-template-columns: 1.8em 1fr;
```

```css
  position: relative;
}
#enhanced-protection-message { border-radius: 4px;
font-size: 1em;
margin-top: 32px;
padding: 10px 5px; }
#enhanced-protection-message label { display: grid;
grid-template-columns: 2.5em 1fr; position: relative;
}
#enhanced-protection-message div { margin: 0.5em;
}
#enhanced-protection-message .icon { height: 1.5em;
vertical-align: middle;
width: 1.5em;
}
.nav-wrapper { margin-top: 51px;
}
.nav-wrapper::after { clear: both; content: '';
display: table; width: 100%;
}
.small-link {
color: var(--small-link-color); font-size: .875em;
}
.checkboxes { flex: 0 0 24px;
}
.checkbox {
--padding: .9em;
background: transparent;
display: block;
height: 1em;
left: -1em;
padding-inline-start: var(--padding); position: absolute;
right: 0;
top: -.5em;
width: 1em; }
.checkbox::after { border: 1px solid white; border-radius: 2px; content: '';
height: 1em;
left: var(--padding); position: absolute;
top: var(--padding); width: 1em;
}
.checkbox::before { background: transparent; border: 2px solid white; border-inline-end-
width: 0; border-top-width: 0; content: '';
height: .2em;
left: calc(.3em + var(--padding)); opacity: 0;
position: absolute;
top: calc(.3em + var(--padding)); transform: rotate(-45deg);
```

```css
  width: .5em;
}
input[type=checkbox]:checked ~ .checkbox::before { opacity: 1;
}
#recurrent-error-message { background: #ededed; border-radius: 4px; margin-bottom:
16px; margin-top: 12px; padding: 12px 16px;
}
.showing-recurrent-error-message #extended-reporting-opt-in { margin-top: 16px;
}
.showing-recurrent-error-message #enhanced-protection-message { margin-top: 16px;
}
@media (max-width: 700px) { .interstitial-wrapper {
padding: 0 10%; }
#error-debugging-info { overflow: auto;
} }
@media (max-width: 420px) { button,
[dir='rtl'] button,
.small-link {
float: none; font-size: .825em; font-weight: 500; margin: 0;
width: 100%;
}
button {
padding: 16px 24px;
}
#details {
margin: 20px 0 20px 0;
}
#details p:not(:first-of-type) { margin-top: 10px;
}
.secondary-button:not(.hidden) { display: block;
margin-top: 20px;
text-align: center;
width: 100%; }
.interstitial-wrapper { padding: 0 5%;
}
#extended-reporting-opt-in { margin-top: 24px;
}
#enhanced-protection-message { margin-top: 24px;
}
.nav-wrapper { margin-top: 30px;
} }
/**
 * Mobile specific styling.
 * Navigation buttons are anchored to the bottom of the screen.
 * Details message replaces the top content in its own scrollable area. */
@media (max-width: 420px) { .nav-wrapper .secondary-button {
```

```css
border: 0;
margin: 16px 0 0; margin-inline-end: 0; padding-bottom: 16px; padding-top: 16px;
} }
/* Fixed nav. */
@media (min-width: 240px) and (max-width: 420px) and
(min-height: 401px),
(min-width: 421px) and (min-height: 240px) and (max-height: 560px) {
body .nav-wrapper {
background: var(--background-color);
bottom: 0;
box-shadow: 0 -12px 24px var(--background-color); left: 0;
margin: 0 auto;
max-width: 736px;
padding-inline-end: 24px;
padding-inline-start: 24px;
position: fixed;
right: 0; width: 100%; z-index: 2;
}
.interstitial-wrapper { max-width: 736px;
}
#details, #main-content {
padding-bottom: 40px; }
#details { padding-top: 5.5vh;
}
button.small-link {
color: var(--google-blue-600);
} }
@media (max-width: 420px) and (orientation: portrait), (max-height: 560px) {
body {
margin: 0 auto;
}
button,
[dir='rtl'] button,
button.small-link,
.nav-wrapper .secondary-button {
font-family: Roboto-Regular,Helvetica; font-size: .933em;
margin: 6px 0;
transform: translatez(0);
}
.nav-wrapper { box-sizing: border-box; padding-bottom: 8px; width: 100%;
} #details {
box-sizing: border-box; height: auto;
margin: 0;
opacity: 1;
transition: opacity 250ms cubic-bezier(0.4, 0, 0.2, 1); }
#details.hidden, #main-content.hidden {
```

```css
height: 0;
opacity: 0; overflow: hidden; padding-bottom: 0; transition: none;
}
h1 {
font-size: 1.5em; margin-bottom: 8px;
}
.icon {
margin-bottom: 5.69vh;
}
.interstitial-wrapper { box-sizing: border-box; margin: 7vh auto 12px; padding: 0 24px;
position: relative;
}
.interstitial-wrapper p { font-size: .95em; line-height: 1.61em; margin-top: 8px;
}
#main-content {
margin: 0;
transition: opacity 100ms cubic-bezier(0.4, 0, 0.2, 1);
}
.small-link { border: 0;
}
.suggested-left > #control-buttons, .suggested-right > #control-buttons {
float: none;
margin: 0; }
}
@media (min-width: 421px) and (min-height: 500px) and (max-height: 560px)
{ .interstitial-wrapper {
margin-top: 10vh; }
}
@media (min-height: 400px) and (orientation:portrait) { .interstitial-wrapper {
margin-bottom: 145px; }
}
@media (min-height: 299px) { .nav-wrapper {
padding-bottom: 16px; }
}
@media (max-height: 560px) and (min-height: 240px) and (orientation:landscape)
{ .extended-reporting-has-checkbox #details {
padding-bottom: 80px; }
}
@media (min-height: 500px) and (max-height: 650px) and (max-width: 414px) and
(orientation: portrait) {
.interstitial-wrapper { margin-top: 7vh;
} }
@media (min-height: 650px) and (max-width: 414px) and (orientation: portrait)
{ .interstitial-wrapper {
margin-top: 10vh; }
}
```

```css
/* Small mobile screens. No fixed nav. */
@media (max-height: 400px) and (orientation: portrait), (max-height: 239px) and
(orientation: landscape), (max-width: 419px) and (max-height: 399px) {
.interstitial-wrapper { display: flex; flex-direction: column; margin-bottom: 0;
}
#details {
flex: 1 1 auto; order: 0;
}
#main-content { flex: 1 1 auto; order: 0;
}
.nav-wrapper {
flex: 0 1 auto; margin-top: 8px; order: 1; padding-inline-end: 0; padding-inline-start: 0;
position: relative; width: 100%;
}
button,
.nav-wrapper .secondary-button {
padding: 16px 24px; }
button.small-link {
color: var(--google-blue-600);
} }
@media (max-width: 239px) and (orientation: portrait) { .nav-wrapper {
padding-inline-end: 0;
padding-inline-start: 0; }
} </style>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.bad-clock .icon {
background-image: -webkit-image-set(
url(data:image/
```
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAYAAABV7bNHAAAFo0lE
QVR4Xu3cS1OTVxwG8Ha6dsZNt/0S7ozX
+wUSGKN7ycIvkJ2OiNcdbvwMfABXLS1VvLXFSMWUgFAh1oJICCEGq8UFp8+fPu87J
4S3vrmcvlfOceaZMKOsnN/8z/
+c95Yv3B8XFxcXFxcXFzNRZ89+rZLJJNKLDCAZdfp0Hini5zWJ/Izk5e+QAf7bpPzf/
yvvKLkkD0I11lArCOqwazL70D65Xdub5RUagcA0kDJCYCJ8Hen5b22D8y5czvVmTN9gCk
RwHzwXvKe8t72wij1JWB6AFMIBDAPVZDPIJ/
Fth7zDWAeESD64LPIZ7ICBwAxZBFRlmURiUWLk0ymuCwrG8NtQyqKKfUVYG4RwP7
gs8pnbhsOAO4gapvlTluQzFaO+Uoy3nNMAayurlaIcOqUKaSUudUKTa9dQN/
t3m0EiWOItX6fw6W8nUCDyFJHh5ktAMbUuh0yN4HtBPpWgGlxNYYQUTSBhTC3Zccv
W3QNoawUR6Afk7p49ZpAwtuYPPHlsFUkFEehHAA0jy61GwthkjM1UTx8BIq2gIeDcQ
+7v3atKnZ2trqK+xs/n8JRFVBUEJL+CBGgYQD/
t369WWWomEMTZ0PgkAaQ8g6ik2RJyH+/
apxwD6BSm3tpLS9QPxTGDUFTQkTZrT6wGApIJGGkKcHD6p38XjLzkzWfw6ZADb0oHv
AEaBHAPpZKohAv7YSCWOup3r6CRBpBX3v9R
+pHuSxAB04oDKI4EjGDh1qDRLGXA9Q1iagYQFigxagJwR6BpwxptIsEsYc

+roVANajBhpkcxag+3qDBs5TRMd5jowfPtws0rqMPUz1JAkQbQ8i0F1vBWP/
GQHOqEwvgWGywPmNWW0GCWMPA9RrA9Cg339YPQR6AhwBGtuEM47kjhxRE0jD
SBh7GKABC4CqGzSBZHplNCDA6PGBXhw9qt4nEo0ADYQBytgCJLvnB97yLs2ZQM
+Q59WV4+NM8nWqESSMPcwOOm8BEA8v2KC5QcxIgxYgTq0sgSY8HFaP4Ewjvx87
Vi9SPgxQ0QYgHqD6q9cIK2jU7z2sHIY4NUAzyF/hkYphptiaDUBjFy6oYa//
aLtnArHn1FaOjvMSkdfZsEgY+7YBqpTLKnvxonrI/jPiTS/
ijBPHAxKYFxrONDLD5I8f33j9AKQQQHZOsaVr1wSlBmkcSN7hxagGlPtM5Xg4swR6xX
wGqWhnk2aWr1/fEmni0iUfiNPLX84nQ+L8wbw+ceK/
kPKGlnnzSFNA8nbPOQBNalOKOEw1DiMwfuaQj11dDS7z3ChGmVIA0nRvr8oCaKPvb
MJ5ycxqOLObKudP4syfPCmpRcLYDRxqmMlKANLM5cvSf/
zqmUI4rQik9ZwAnDnkDbKA/
K0jYeyhD1ZtRnoFJKxSWuVwWjHEERjiyCsrR8N5iyu3i4iPhLEbON1hNuUApNdXrgAoo
HJqgHQchDiSApE
+dXXxdIeBE2am8+7GjS2R5oBU1ZCDp5WfBR2HweVtueaWNXDKNXqkeSARRlutiMP
404pZJJCHU0RwKanf8El786kEIL0BEisnsOfoOJICIzjLSLmjY5fFl33CZ/
XmzSCkwMqZD5hWgkOgnIELh/
Yhvb16Vc0RqXa1Ympx1EoikTZw6TnavA9AWujpERgdh0B+5bAp
+72npBKJHQZuXrAPqXD7dlDlEIdVw
+BGCKmePgO3v9iHBJzaymF0nCUNB9f3C6q7e6eBG6jsyofz57V9DrOpIRc0GMkKUun
s7DFwC56dwYGnvpTX9hzGwynF47wFz8BNnLYGx1ScUn64WlXjlOPxxY8Yk6HbgO1H
0nsOo+OsVRKJmNEbyW3PGpAAhGg9hwFOyj2KQCR/
WjG4TeaWe5hlE5LAlBG8bvUwi3sc6lN390blGMBxD9S5RzlNxD3U6x4Ld18s4L6awt64
LzdxX4/zDxj9/IEueAvhAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAACvl
BMVEUAAAD/gID/gID/VVX/VVX/Tk7/YmL/YGD/VVXzUVH/XV32UIL/W1v2T0//
WFj3UIL/UIL3UFD/WFjwTk7/U1P/U1PxTU3/V1fyT0//VFTzTk7/UILwTU3/VVX0UFD/
VFT/VFT1Tk7/VVX/VFT/U1PyT0//VFT/U1PxTEz/UILuS0v/U1P/UIL/VFT0T0//U1P0Tk7/
VFT/U1PuTU3/UILzTU3/U1P/U1PwTEz/UIL/U1PvTU3/U1P/U1PxTU3/U1PzTk70Tk7/
U1PyTk7/U1PzTk7/U1P/U1P6UFD/UILzTk7/U1P/U1PyTk7/U1PtTEz/UILyTU3/U1P/UIL/
UILxTk7/UILvTEz/U1PvTU3/U1P/U1P/UILxTEzxTU3zTU3/UlK7Ozu8Ozu8PDy9PDy
+PDy+PT2/PT3APDzAPT3BPT3BPj7CPT3CPj7DPT3DPj7EPj7EPz/FPj7FPz/
GPj7GPz/HPz/HQEDIPz/IQEDJPz/
JQEDKQEDKQUHLQEDLQUHMQEDMQUHNQUHNQkLOQUHOQkLOZWXPQUHPQk
LPZWXQQkLRQkLRQ0PSQkLSQ0PSZmbTQ0PTZmbUQ0PURETVQ0PVRETVaGjWR
ETWRUXXRETXRUXXaGjYRUXZRUXZaGjaRUXaRkbaaWnbRUXbRkbbaWncRkbdRk
bdaWneRkbeR0ffRkbfR0ffa2vgR0fga2vhR0fhSEjha2viR0fiSEjia2vjSEjjbGzkSEjkSUnkb
GzlSEjlSUnlbGzmSUnmbGznSUnnSkroSkrobW3pSkrqSkrqS0vqi4vrS0vriYnri4vsS0vsi
YntS0vtTEzuTEzvTEzwTEzwTU3w6OjxTU3x6OjyTU3y6Ojy6eny8Lz8/
P0Tk71Tk72Tk72cnL3T0/3cnL4T0/4cnL5T0/5c3P6T0/7UFD8UFD9UFD/
UlJJWZWgAAAAYXRSTlMAAgQGDA0NEBUWFhwcHR0fHyAgNDQ3ODg9PT4+QkJD
Q0lLS15fdHR1fHyEhIWGiIiJiYuVlaioqaurrK+vuLm5u7u7wsLExMXGxszM0tTU2dna2t/
p7Ozt7fPz+fv+/v7+jD+tjQAACYhJREFUeAHs1cFqwjAcx/
G1FR0iIqKIFFFEUHKJQKIIRFKGUiISKVVCmiHrKpCDuPHcbA99xtjA1+b7HLjmMkaeIu
+TzBl18C/xtFURRFURSFQ6bc6g0ns8Uq3u3i1WI2GfZa5cz/tKQafW
+NX629fiN11Rij1p3v8af9vFszrpRTGWxAZTOoyK8pdpZgsOwUpeaUxgSMyLgkLac6BZ

dpVUpO3QM3ry48JzdCIqOc0By9vUVC27YurscMIUBoiprHIhCCWEJGyvsQxhfwk5oxBI
qbCXM0m0AoYmuJrqgD4RyDvyftQgl3zduTDSBFcMvXU4ggSVTg2ieCNFGW4/8EkCh
g/keGC6lcg61Hc8Dg7cc76DkaU5ANnqAnliKb6V4QrqB7liLCcEXyMfiCHp4/QC3O0/
boPniDDi8MRb5OGWSBN+hwOL4yFFl0PSYB/
0LH0+UTtIhJ9WAhEgSdTufLF2iFNl92ByRa6PzlUPTNih21uFG1cQC/
Kewn6GWh0BdeSu1NKRRKS2+EUuiF2CKl7UWR3WabNHFmk84k2WSTndmsGyfZMc
42ziYmJhozbo0hxSVsMYRYFhEEFWt7Uau1V3q+hf+Ts2ZXMsmMnHk+wY//ec55zjn/d/
Heuc4DmqUg/4pr0XXnO+1Zwp2QP5B1LTrr
+D4lHKC3aQ8BFHxn3bXI6U37OuFPKBAUI4U/3b6yHf4TCIqrh3w
+vwCQVHQrmv4TcYFwJ3THHxIkSYqZLkUXpv7/
EC4QTmrf3J1QSAwDtFh2KZr2f3SOGzQXCNwNCbIkx5LpqjvRuSn/dbcIbw/
5fYGQIN6T5EQ6rX7qSnRr8q/fcYLi2vaz876AIIr34olESlEzzb+Iizo+EfQmJwie
+WAQLXQvGk
+mlEwma7kRvTHxv5dwJ3R7PiAIoixHk8mUmnErmvR7fJoXNHsbp3RQkMLRaHw5BY6
maV+4EJ2e8PK5xg2apaCIiJZOJJfVbDarZXNtZ9E1+zfREclNoi0UXJBjMbpiq/
BouZzWcRYdsQWd4AfNYY4tLOAQiidVdVWjnpyuf
+UoOmELusQP8gfQ0wuynEBCqwDldHh0w1F0yc5z6CY3aM6POSZIsryYSCtoIayXXsg
bht51EN08ZAM6SrhBPl9IFDDpFxfT6QxaGvkUUEbR6DuIjtqATvGD/
L67oihF5MTSUlrNaDoKHN0oFs2vydQ6ZXs141+yu4IQFumeX1LX1vKaRj2GUQSo9M1
/vqZd9SAhP0BhKYEOUjJrWr6gF3Tq2TBNs/SYTKmr454Zwg+ax6AXo/
F4Mqkqa3l4DJYPOGalskum1MwY6LAHIAQkShIGq6KghfI6Wy8EVCqVKpXyt2RyHR4
D/c8DUCAgRiJR3Dxw9UBLG6yfTXjMUqVarX1HJtaxMdBJD0BBQYhEpEQqqWBs5Kjl
+CeeKkDTRCfHQGc8ANGAZDmeSuGYph7WP0UmqtUatcb3ZEKd4X9v2CUUBAhjQ1l
ZwdgoIB54hhzkU6k1GvXmD67fHpc8AAnhSCQmJxQEIF1f3yiAM/
SUwUFA9Uaz2fzR7TS77AVIACiWximEMVYoDNeL5VODh3KshmUvujwGuuIBKBSOyA
Ap6iruHRs4DunuMoftzECWtbVl/URs6soY6C0PQO+KmPRoIUx6BASQud8/
aCDLagLUav1s9wM6BrrhAejJe1EJPa2q2bymb2zQfEbx1Fk
+8LTaNqIbXoJePt8XxZMJVVWyGKvY88P1QjtXWf8AZG21HrTbrV
+cQTxL9ur5vuj9ZAoLltHzBlasshcQPA14miyfdrvTeeawZJxN/
fuTA6LllJrVADKKFRQ81RryoR29hQKo1elsbz9zbOrLxCPRB
+oqNplxv1iulPfyoZrhBoMG+WwD1P3VadtfJF6JPqRzo3Afoio7EJt1ixaLp418HnZR/
xZdHAOdJ96JNGx6c7NC46Ee1Kh/Oggl+XS3d3ZekAN13nm4cog
+Wsce29xkGx7FNrw1BIHzcHunu7PTe/Ri6nA9SbwTPf14vWCW2X5nG77J8kE/
dxAPDajX6/V/l6N6bQx0jHgp
+sQslsusf5qsfSiIctoUNOT0+oN90THuK6yD6DNcERusfxoW2+5DD2sfgPr9/mDw
+OXkK+wM8Vb0eam6Ny8sxnkAUBugIQee3mAw6I9EM/
bPIG7R0wOien00MFp0YAw5XfRzr9vrPRr0wdnd3X016RmEh6IH9ccB0ZeUM1qvv7s53
9YmgiCMx5S
+KP4RFUREBKFaFSqiSLWCVVSwBdEKIr5QP6IgtYrYiNe7unpudleWpHlrxdRv4cwOm
yYVmSudg6XzCX7MPnN5sjsz+P0BoCXIz+DAsizPikB0T
+avNEP0mvwP8dD3Bw5smWgQBzJUFCFHV5jLBiGiN+R/IPD3Ar/
PpGcssCz7nCNPodQGXTYw1zEiRG/j95n0PBB00DPyKOAp9QZdx/
A2XyBH5H9IPwCEcg4FFs9LlWtaqwfMlZ4g0TvSc8BBQbeiniE/GeSnhNDmotylJ09E
+UEBYX0BzUfAQQGhfABIK21OyV0L80QfyP+8h1hGnhbVF
+AUOfJobZ6NSV6c80Tkf0jP+H2m+qIEKeAxVyWeFpj4uUXUXQk/YLG
+SM5ZEWhK5DHHG/+Jx4JEv4ZytEL+B3FIPwWEWqPzMuaR8PMUT9T9tBR/

UGO5F0E+CnisPS/zgMcTdbeIVsn/oH5Az3mQDyWobV8cEX/i5InWV1sD/
WR4XsRjMEE3+UfgWojI/
2SxvEJ9tYHHHZN7JueJ1iNQ7yv5n3BeOeGUxgLPrRoaCXii3rfof0J
+FMonCNq4E7KtFjwR8Qz5nyhnAzTO3ZVvRuGJel9G/
A8dmGnjgbnTDSZuyBNhfkb8T6mNBhxnnb8u39DEE33/x/9oAwFE/vn+Bhvn/
kjH73y7/9EWcaz3Z+tpiuOiX2zzP8biiXn/sCnfNliRaMT/AJBzwPPypHxjZVUiNex/
qL6cvyTfelqdqBzyP6Sfzv1mXc25lYj0wP+0rYPwTw/W1r5cjchE/wM8oOhXZ
+pr8K5OpCk9cF7+co0t8JWJgn5MILq2L4EhgT7yBBx/eyyJMYq+Czw/OgvjiQya9G3Iz
+JEMqM4/Y71ncXDCQ0rbXb8wkRS41ybd8bTH3hLfyQw/aHJusZKDyU/
eLsHRpMpSVO7H96easqOt8/ujmf2wF5fAJDgioQEl0jQmo0deYAnF46mv4gk/
VUtKS2zkV/3Mzk9Mzcf1/3Mz81MTwqt+/kLc5W5R5JoGz0AAAAASUVORK5CYII=)
2x); }
</style>
<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.captive-portal .icon { background-image: -webkit-image-set(
url(data:image/ png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAQAAAD/
5HvMAAAEyEIEQVR4Xu3afWhVZRwH8LvdmjVSjKyQbKazia1k1sL6YzXoZdy28/
J8f5XRC7hwEVk0hZQCXY2SipCgYgappXMQ5Ya9ELZqoEM3VHqxBkapm21jOBWmb
Lfudk+0Peeu2+mc3zk9514WbN9/zz18ds75Pc/vPM+JTP9NZzrT
+b9Fn2nGxHrainb8ggEaRgLn0IOjaMFroU/
Ssg1yxBLZpxBD00QufpODrRiFX6IiUKXY8XcRAJsryC3nRL7EpsQJ/r0V3iSS0/
MKU4D6uoQwLY/P1q4i0aYX9xVqyPRH1jbr8UtfSbE8CDqIK60wBjOEb7aDf24CBO/
gN12LjJF4dM
+dPAoNgMdNkUxLENQltAFViJdbTWfFTcaRbiMXyCpE1CM4vR5uAzdwB/
hcxCDI5jXqGFeAadNJoGH0ErVhg3YPs46Ad9JldNS3GCAbDPkFGG/
eJGWktn3PA4IUB3oxPzGY4AXSBLCSTLgfZxx2MLV1Q5tFHeWxWQDNalHuIWeo4q9
GW4VVSnng5J0KgZ8+bkYpcEqIJk6H2y0CiW4H58gGMYQhyn8DlqUUltZIk1XF01kKU
W5yhm3gVB3Q76MDaRyXBQR1aIlJnyi6jNZUgs8uTQfUiGDJIxZ9OPfxEwhAO0F70To5
NR5skxrsUgWaGDZLQCHBHVxnKxWjwuivRS7McK7na1kpUZkEwUzXbFiZfZcVk8pE6
xZ3u34AV5TFUsV2ObBHU0+Q5wu0eJ21krw
+eD4lzBvOYf7wLeyI5DEfLx2l1DPqoSXK8SItis/gm41mlago/
9P2UApklZE0pkHhdAv6YIiB8NwEQL9HvUwCkzZHz15g5Gw8jmR0QtsizvOoACUjAIdl
OZQUkHpCgDqd1gwRslhX3djZAmDsBwpATJPtDUZPqqGFuyUWU453lWu7fFHZOvhjiQ
+aqiDheQ/bJjXpf2oP+caRA
+cgPJpkxckeZfilHMgmirC8gee4rz0qF6KV3IJEi86R8kg0oazQ6Iv2Uyoiajt4x/
qJ1BfcCe6JQZC/BQ82XvDLYHJPUEK3tmYHTmlotpbzCSwsDomDpc
+mD6VgHETB3M5Or2c5xUBvGTq7P98DjBEjqrBOLbD2eDZj7o+aJdhnhooJxUbd/r1cJ
+yl1mJMMBGWUSP6zlezT5SlirGNKacEB4bwKEXcxrEN7gThUGSCtILaZXMC+KiGN
+5kHYTTndkSj7Ko3GTIPMEvtJFKt9LDYgiZtVQTLnqUPUOJcV8KWE95df4m85ps0+CQ
/ig68xN72lscF4yveCFTa5g9AXnDR5lczC1Jr
+4UhuCsAv6YlH3EDURFbgPCG3Y2altmPG9FlJ8LnoOYLbXNDzgi
+Myvkqii9sIuocAHZZuN9tAKi6hnaiBwka9f942y2rzFcut4tZOO
+3r5J6xcVmYEeKc9xlPuC3FhDHynBA6LQ5GGA2ftnNI82RqDrIDgb1ZU4AnxzUTc7raD
eWhwPCUW0BA/C7gYeP9cWqILSUX8YAgmxxloF3tYL/DkK9Y/znwm8CI4lvRLV

+dVAQThOpOPht8p+ogZ4W9/jijFFD5eXhWPgPCXjMh/
pSVYD6pxaywPGOKOIBWfkYhXYKozgvy5/
rYBva8avzcx19sSrgT2jJnjJMVMiUAAAAAEIFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAQAAABNTyozAAAKSUlE
QVR4Xu2de2wUxx3Hfxhj5dkSIgOhpICDWilH1AjTGEwdnaCNdbi+2/n
+EGlKAvSBopRlKa+KSC5tCU
+ZoKaKQ9WSNihqgsXDqRMkHBRBDDi04Q9kGQFG4JAEqF1TIkwSP
+BcjBXK7zC3551Ze3zdz/7NnffD3uz8HjND/
hAQEBAQEBAQEBAQEBAQEOAMjeWpOVjF5djFB7gWDWjmNm5DMxq4lg9gF
5djlZoTy3OGkmceuzNWoBaiDDu5Dmf4M26/en2GM1yHnShTC2MFj91JNhG
+y4lwKVejkTtTv9DI1VzqRMJ3UaoMxlS1Bof5iuunX8FhtQZTaTD1J6EsNU2tRA06uNP7
hQ7UkCuxB3kDN/X605t4Q+xB6g8wGWV8njvNXJQUnoH9Op+O/TyD
+g5nLEpwXAjwURCK+JCR7ziEIvKf6CRUIC4E+CiIc7DT5LdgJ+eQf8QKuEoI8FdQBi/
jL41/05e8jDLIPM50rhYCfBaE+/Cei4DL+Ce2ogy/4We4i2fwG5RhK3/
Il5NLwnu4j0yC0dgqBfgvKKmeI/wSopGv0S34/tcRxR/
4SDJFZIpwplrCLdzZ54LG9DinalG/Lx5H1+AQP4dNqOSDfIiovXb1O8UFUYhM/x6GvRjB
+iS/1NP/
CGDIDHuU6DQFaY5CaglZxW2d5WdE9RMV3qDn8Op9LKvwcv67mTLmdqOgePI+zQk
+rmmLo2eG1Ou8r/beY8+R1AW14PpRFFMlWv0NzqtLxb/
w2kk0UysLz3PaVIOdJMkHsfo2pmbHXPFZ1z2A4RPTDb/BG/qLX6r/
AK9FRRBzqnk1hFZkAReL/qd8E0SC8iZJwJhGewgXP+v/T9dSEM1GCN2kQaZOBdTo/
LfOhhhqOCt3Pxw413FD4iS1CgAWCsNvEN2A36RO9W/wxlggq/uYt0yjH
+DW1Ri1Us69eC9Ua/JWPCS1mX
+1quAgLrRFEFMvnNnG7cbyDeRh9i2ntPLyTMEi0YSrpgjGo5057BEkw74ab/
Ut3hieSjZn8MvbyEW7idm7iI9jLL2NmJPta9ujV/
0IVPzHw9Ag9VgiScGl3Zsd5gCh3iPMEasRTIp+vGvWjcKbzAPZ16cF6A2OP
+HFZKYgyeLtaQhmUwc/ypyII/5SfDWdiMW+nDO03lxiaLRMk4RA
+SF07PuAQaZMhXuyWC8KOXqrfTrpgHXfaLUgWfHo1GBzSLv
+gCHHbBUlmjMRHqenBRzNGaoekIuayWJAsAN0Ql7VjL1bgaWcWnsYLeJ/br
+u5oF3uCWeKiN1qQRIn3DXHQTN
+XXQPCQqH8fJrxag2J0y68FohwHJBEjUblWo4USwfr
+Aot3ALjqKsOxUWHcFvq9kGsoWIWyPoEnmgS0SiaFRGR5jKF9bZ8/
zwKfIA3u9JNvaSCXipRXo6sYM8oCb0kJK/
pCYYKeRwiw1idlNJPH6T6sfJBNhmkx6cLb6DPKJeFKJfNFUl7bTpUj8l7wzmPdcF7THU
DcTVNunBJtliko1PuvTgk0i2oRYEm/Twn3KHkCbOI2hFq/MImYGrbJGDj/EUGUHNV/
ON9fdYoKYF9diCH4dvI/
tAhQWhg704YxEPBCUBJckFBlLqA0FJwOTkAgJBZYGgpMUdPh8ISoKallxAIGilbM7Hv
kCQADVCwKLCYXwsEHTDgiW5Isd5+Nq08V/pJ8h5WK4kSnGxlRMR/
6y5u1fPyeVL6SaIBslanxOhVOBSIWjbDQuNOqwWpJ8vLfWQJFML6Dr4eboJUguEoGp
KBdnlJzP/WGFaDy70q6AJsjsxpaW2cuBKzOLxa4YFvU/
9SO4QOWyksHw4licE1ffwke8aFNSG79oUlMfyyA01V9zA2z02AB82Iucc/
s4h6mdkSVrNITewWjxB66kHoqPwcboEElgv7ne1u9FyYXT+Lfv/
LqSHIDVf3EF5bysZhcn6btJBEBcKQVW9jMOSLShznkA8DZ6gKWJIqXE3WitG9YcoCfj
VwBcUe0jcQS25gQYRnYzVzTzaX70RT1ADuSHDN3Wva0/7WwNbkLo3lTR3Qw69oSx
yYcrtfHAgCwplyYmrIUGyZ4JPpJkg/Z+YJDqem9LvJ6Y/SMsI7os0G6T1X/MSFeMr/

xeveT4gBOVTyuAXPheIzmEr5xgXlC8EHSA3sEuGGpb147eob/kZamCXTrDqziB
+w29FqPQ3WNVNd7i/
NPf4rKjJ13THKnejc1wSZq4pW78XLvibMPOQcvWwsuyMrYLcU64aSfvUUd/
BxYEhKHcIX5ZJey9lHw+oH3C7HYL0yz7uhUNPqLlWCNIvHLqXnm1rA/Wz9KzRvOAF/
NkCQfrNC+7tLxpLgXdalEi3/
cW9gYo8IBb3WyqIF8mEvdcWvH2kQXQEGmwVJFsL1UrvTZw5Woq
+zedtFMQ5Urya5r0NeDlpob7HX1ooaLkQdD6U5b2R/
ARpgpl8xTZBMpOOMq2lCLF80gS/
tEtQLF9qx2TqHXIfcWwmbewShM1Cz3HNeTA6iselk6DicXKuhxL9BXUb00kQ/
ij0xJ2x2ksy0RodlS6CoqPQKgRVmFnUuyFdBPEGqTw6yciycHyOMekgCGMSipxV5jYW
eIs08G3h+HF+w5mV+jQvsR8lVmBwawoU2SNIXjjJTCmAogTB1WY3NznlfYm//
6sYsd5tJ83wbXxKCnKmkx6Jx0BgBXkE//
BbkHsdDy9IPdhqfIMldHg4KUCkcn2+OFnDJjrk
+IXRpI9aIq3z6cJh5AEnty8E4aQoVMltAU9LmWqJ0U3e9OvjeLcvFDmzqEdQmSCzLpx
JAoPbBGKxtyoUWv0XhL9RD2Cx1IM4HiWB2Y0m272N/
mp2Hwg63uPbOLGQuZYExrcqxUVMJA/gZ9zudw8RJYCJiaVw7Hf9eelvdovG6HiP
+4J86K+ixBZTNEo9aI7dT+7ob5eMkx43GR7kRPhVHMVFvsKX/
RU0Y2Ti1BBxl3jA6IbbtUKRJSGI0FObqA/ryAWjW7bjZHS8rYKi43HyJj1bXMIR85v
+oxETbRSEiWi8Sc9ut6jfl2MjcNGZbpsgZ3oPTVyHoneTBjoHj7TzIpsE8aKbJxOo1zr1Sf/
oGlQWDrND0PWgQujRyIcaO/zotJqiIcjP65DG02Py+Cx0YEX4NtsEYbfO2GP
+ALZTKLJJELbov7nMH+H3FsbYIAhxrNOY9/h5CCQ+5w3RUf0rCM26QYXPx4iilTcWj
+svQdhvIiT1/SBadGBzLL+vBSHOa00mNPw/yvgEL
+ecPhNUJ7OF9jxHS5MfMYF9vMh5mAb5KqiFl4YzyVYwGttSGDq3qQVqQu4Q84KwDa
PJDvQP5EcH6vltrDemp1qEzHYTK+jjAwOqYgU00IhOQoX/
274jjgrR3zPgdsYvQb1vcupR4oylgQ8mo8xwb8d5lGEypROhLDVNrUSN3nan6ECNWq
mmhbIoXQnf5US4lKvR2CsxjVzNpU5ELFhKd5yhsTw1F6u5nKtQw7VoQDO3cRua0cC1
qOEqLsdqNTeW5wwla/gvpXzJeo7GTncAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .captive-portal .icon {
filter: invert(1); }
} </style>
<style>/* Copyright 2021 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.https-only #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.https-only .icon {
background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/glDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzZXXRD
XXQzbXQzbWQjXYSDvWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP
RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLIPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////

PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTlMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnUFkRXwLiK78llgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPIElTpIRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3DiHUhftKIlzdKoRSFe0r8sXDAkSoumkIigYaIOkIf
eWi56EESFm8r1w0fFIl4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRI/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXiz0987BBtux65qaCeF8lHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7lcfH9E3ZlEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9rkJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKFtf+huwrjaa6Pxv7RfgW7nubJPtKl/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUIEQVR4XuzWS4rCQBSG0euz56ISgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMlL
5W6t4Zkm22Ikm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjITvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYI0gGFkvaR779KKZUWuFKe7nlIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGI8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n

McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjIMbDK01U0OVZxcplE6BIjLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyleiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFI83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiljRDx
KQwJk9a+o2u431gYZERBo/kcBlfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzlHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPI
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
filter: grayscale(1); }
@media (prefers-color-scheme: dark) { .https-only .icon {
filter: invert(1) grayscale(1); }
} </style>
<style>/* Copyright 2020 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.insecure-form #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.insecure-form .icon { background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAYAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqWxeWyLjRH60BYpKZHYpoFC
U60/
xKCt5ME3QaSpT6WUPElCEXyTUpIojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2
WZPefO7LjrzjYzcJmZc8495zvf3l3Zu
+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7ivo1Ssrm5WYL9AZSC7OzsAuyzl
HuCHcsjyOawZ7lbVFT0W09Pzz843yz843Pzz843
+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGcvr7+2Pr6+j
kgOuILVeKVJnjJzc78eGBi4nXhVve42kEtTY2Gci4nXhVe42kEtTY2Fi8vLz8/Pb+/
r65pMVIWKwkEodV8vLGx8SPY2 Be8scH78gKTFnJyc02hN1/3Ud9/3Ud9JZUtVx1i70fcDti3YR/O (truncated)
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisdpXKAWJeLi4tburu7/1VMXMW
+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqIPo975FrGbEytV

8QT
+JlnTMT0vyRRD6nEsAZLutOlpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXFf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8ln1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYlaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xlpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVIGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6Nol
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+lVbEHR3pcBg
+zgXEki6IMuImdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusClDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwljMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnElXSzlth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYlhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yluhn8Aw+70ZTSym/lX0zRiMxZiaJ2llZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz
+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh
+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWlBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAYAAADnRuK4AAAAAXN
SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxQSZTsw25wAUPiNQTRgFfkv
8YlbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENFZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39aJT5/
zPv6urq6q7m5rCEhAICAQEAgIBgYBAICAQEAgIBgYBAICAQEA
gIBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44d0/
a85ubmyWnnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTPryeeeeaZX65v/5/HI

+pZUwQ6I477vjD0NDQAgiwgOBfynYa23E+I43OY+jcy/Zjtn0tLS19zz///Oc
+y8ijroYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGQF1dXRP2799/
G2TpBLCbWFuyBM5B9xB5XoVIPVOnTn2xu7v7sIOO3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPiIvWRWrcEWrx48XyI8xA1znX1AXVxK6mR3oBI
qzdv3qxbXd0tdUcgapybIY2IM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtbLcoeiR4
Mg0UesXRDpPY9qM1OVewItW7asjT6bJ0DgL6y5t9dTpI6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4IUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUMa+2o6/fTTJ//yyy+Tjx49OjxOhsxFJA+PobE/
PJ5G3kmSrcLyZFtb2wNr1qw5UoWyKhaRCwItWbLkIsaqthCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFFjjupOyYaecbt3756Nbo21acztGraZEQr97zPW1vHcc899dYohN
TioOYFo78ygvfMavl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp2bNnzzz86oRI7RSo0X3fyz78uoF20R7fii36akqgqG/
nZUA+12J0JVII8zrr08htA+BDIeSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqIfQuwgy
+XqA+ZmaaDEkqkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/
JEpCFItBwSqeujqkvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26nAvgKXy43INKkej2atdEvqrRRP6rzRPdtlKRB9APANa
9s2bNuqpeySPAZLt8kC/yKRGIpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axAnzUsJ5HTVSVp7OqEEj9PNzz3wWYNI/
qqqIfZt7MEwCUy3GhNlFXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cl9amSvkzv4Wph1mdhBiShjzq85jPVfV4o5JHgZJv8lG
+cpgm+BcePny4V9hLb5ZL5gTS8ARXVpoe5k8B9AqA/
VeWQORJt3yVz9jk3B0hzKOhoUxdy/QWpsE/+j1edPWAK/It1oUA
+qOrjnrOR7vxLIiwnfVaVz/oF7uN2/5Lrvkr5cusBsL5adzL11cyoNR5iLNt0qRJN45V8ggX
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQIwFg7xuKbnqMDKQyEhSuJFIMoFpncbTlhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4YzI1i4pi9dDbvrIzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjElwe8eOXE0b0+djxWgNxhWio4h0Ms
+pVJ6H6eWr2qM64IKIzkmEIq48+4jWsA5yvBuedHLQYlR4H57ng7O2VIa81EA22bhwy
A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhlxYspnUrDcIeGJ0pLhDPFEIi
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AIDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC
WnIEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqeYbGoCYcjKYoorpD1Tzz
MJK/hW9dMRls9YC3aM3SnFpCKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qkIhJplJ1Ul29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/
Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujIW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc

9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0lhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6lutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrlyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMlOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYlb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMlLmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNlsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnIZDpNkahGpX/
U2HhYT8TBISxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7le
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqlMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTlg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAleC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9EIg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UIgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xlvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrISZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY
+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYlPK49jRrMHC7yyf3vaKSQ2XCR
Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGIYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh
+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1lY9SS3ru4U
+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxIkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/
DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFIGWkt5aeM64/
zv+MLwSXf/

lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0XxfrrdQtRNaq
+IsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOIctHrF217Nux4Fk3BIqdgkg3Q6KHWF0nqc
WqcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tlME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodIQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .insecure-form .icon {
filter: invert(1); }
} </style>
<style>/* Copyright 2019 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.lookalike-url #main-content a { color: var(--google-blue-700); text-decoration: none;
}
.lookalike-url .icon {
background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAYAAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqWxeWyLjRH60BYpKZHYpoFC
U60/
xKCt5ME3QaSpT6WUPEICEXyTUpIojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2
WZPefO7LjzjYzcJmZc8495zvf3Ll3/3Zu
+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7ivo1Ssrm5WYL9AZSC7OzsAuyzl
HuCHcsjyOawZ7lbVFT0W09Pzz843ivo1Ssrm5WYL9AZSC7OzsAuyzlHuCHcsjyOawZ
7lbVFT0W09Pzz843ivo1Ssrm5WYL9AZSC7OzsAuyzlHuCHcsjyOawZ7lbVFT0W09Pzz
843rNtwhqaGh4ZXV1tQGYL9AZSC7OzsAuyzlHuCHcsjyOawZ7lbVFT0W09Pzz843
+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGovr7+eGBi4nXhVe42kEtTY2Fi8vLz8eGBi4nXhVe42kEtY2Fi8vYz8eGBi4nXhVe42kEtTY2Fi8vYz8eGBi4nXhVe42kEtY2Fi8vLz8HVrMKXvY1GjRmvz8/Pb+/
kgOuILVeKVJnJzc78eGBi4nXhVe42kEtTY2Fi8vYz8eGBi4nXhVe42kEtY2Fi8vLz8eGBi4nXhVe42kEtTY2Fi8vLz8HVrMKXvY1GjRmvz8/Pb+/

r65pMVIWkEodV8vLGx8SPI2Z8scH78gKTFnJyc02hN1/3Ud9ZJCkG1tbVfwnEnyMlxBp
DOkcQybG9ifwv6OezvRyKRv5eWljhyZeG4AMcvweYNnHKkq4TNcezzqXfbYLsBm46h
oaELbrZu+l0R1Nra+vz8/HwPgH/uFgj6xwA
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4tburu7/1VMXMW
+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqlPo975FrGbEytV
8QT
+JlnTMT0vyRRD6nEsAZLutOlpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXFf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8In1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYIaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xlpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVIGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6Nol
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+lVbEHR3pcBg
+zgXEki6IMulmdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusCIDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwIjMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnEIXSzlth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYIhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yluhn8Aw+70ZTSym/lX0zRiMxZiaJ2llZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz
+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh
+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWlBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAYAAADnRuK4AAAAAXN

SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxQSZTsw25wAUPiNQTRgFkv
8YIbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENFZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39alT5/
zPv6urq6q7m5rCEhAICAQEAgIBgYBAQCAgEBAICAQEAgIBgYBAQCAgEBAICAQEA
gIBgYBAQCAgEBBoTASaG9Ot8l6tWLFi4sGDB3+P1HStx44/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTPryeeeaZX65fv/5/HI
+pZUwQ6I477vjD0NDQAgiwgOBfynYa23E+I43OY+jcy/Zjtn0tLS19zz///Oc
+y8ijroYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGQF1dXP2799/
G2TpBLCbWFuyBM5B9xB5XoVIPVOnTn2xu7v7sIOO3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPilvWRWrcEWrx48XyI8xA1znX1AXVxK6mR3oBl
qzdv3qxbXd0tdUcgapyblY2IM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtbLcoeiR4
Mg0UesXRDpPY9qM1OVewltW7asjT6bJ0DgL6y5t9dTpI6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4lUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUMa+2o6/fTTJ//yyy+Tjx49OjxOhsxFJA+PobE/
PJ5G3kmSrcLyZFtb2wNr1qw5UoWyKhaRCwltWbLkIsaqthCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFjjupOyYaecbt3756Nbo21acztGraZEQr97zPW1vHcc899YohN
TioOYFo78ygvfMavl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp2bNnzzz86oRI7RSo0X3fyz78uoF20R7fii36akqgqG/
nZUA+12J0JVll8zrr08htA+BDIeSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGIMhzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqIfQuwgy
+XqA+ZmaaDEkqkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/
JEpCFItBwSqeujqkvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26nAvgKXy43INKkkej2atdEvqrRRP6rzRPdtlKRB9APANa
9s2bNuqpeySPAZLt8kC/yKRGIpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axAnzUsJ5HTVSVp7OqEEj9PNzz3wWYNI/
qqqIfZt7MEwCUy3GhNIFXXsjTTG/z/
dQkj3KYppbeN3HixDkbN27cl9amSvkzv4VPhh1mdhBiShjzq85jPVfV4o5JHgZJv8lG
+cpgm+BcePny4V9hLb5ZL5gTS8ARXVpoe5k8B9AqA/
VeWQORJt3yVvz9jk3B0hzKOhoUxdy/QWpsE/+j1edPWAK/It1oUA
+qOrjnrOR7vxLIiwnfVaVz/oF7uN2/5Lrvkr5cusBsL5adzL11cyoNR5iLNt0qRJN45V8ggX
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQlwFg7xuKbnqMDKQyEhSuJFMoFpncbTlhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4YzI1i4pi9dDbvrIzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjEIwe8eOXE0b0+djxWgNxhWio4h0Ms
+pVJ6H6eWr2qM64IKlzkmEIq48+4jWsA5yvBuedHLQYlR4H57ng7O2VIa81EA22bhwy
A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhlxYspnUrDcIeGJ0pLhDPFEli
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AIDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC
WnIEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqqeYbGoCYcjKYoorpD1Tzz

MJK/hW9dMRls9YC3aM3SnFpCKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qkIhJplJ1Ul29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/
Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujIW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc
9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0lhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6lutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrlyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMIOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYIb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMILmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNlsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnIZDpNkahGpX/
U2HhYT8TBISxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7Ie
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqlMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTIg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAleC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9EIg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UIgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xlvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrISZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY
+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYIPK49jRrMHC7yyf3vaKSQ2XCR
Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGlYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh
+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1IY9SS3ru4U

+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxIkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/
DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFlGWkt5aeM64/
zv+MLwSXf/
lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0XxfrrdQtRNaq
+IsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOIctHrF217Nux4Fk3Blqdgkg3Q6KHWF0nqc
WQcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tlME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodIQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwwAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .lookalike-url .icon {
filter: invert(1); }
} </style>
body.safe-browsing {
--google-red-900: rgb(165, 14, 14); --google-red-700: rgb(197, 34, 31); --google-
red-600: rgb(217, 48, 37); --google-red-500: rgb(234, 67, 53); --google-red-100:
rgb(250, 210, 207); --google-red-50: rgb(252, 232, 230); background-color: var(--
google-red-600); color: white;
}
.safe-browsing :-webkit-any(
a, #details, #details-button, #proceed-button, h1, h2, p, .small-link) {
color: white; }

```css
.safe-browsing button { background: white; border-color: white;
color: var(--google-red-700);
}
.safe-browsing button:active {
box-shadow: 0 1px 2px 0 rgba(165, 14, 14, .3),
0 2px 6px 2px rgba(165, 14, 14, .15); }
.safe-browsing button:hover { background: var(--google-red-50);
}
.safe-browsing .secondary-button { background-color: var(--google-red-600); border-
color: var(--google-red-100); color: white;
}
.safe-browsing .secondary-button:active { box-shadow: 0 2px 3px rgba(0, 0, 0, .5);
}
.safe-browsing .secondary-button:hover { background-color: var(--google-red-500);
border-color: var(--google-red-50);
}
.safe-browsing .error-code { display: none;
}
.safe-browsing .icon { background-image: -webkit-image-set(
url(data:image/
```

png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAAABICAMAAABiM0N1AAAA
+VBMVEUAAAD///
//////////////////////9/f3////9/f36+vr8/Pz7+/v7+/v7+/v/////
r6+vn5+fk5OT5+fn19fX4+Pju7u7v7+/o6Ojx8fH09PTy8vLz8/Pj4+P39/fm5ubp6eni4uL8/
Pzw8PDt7e329329vbs7Ozg4ODh4eHe3t7l5eX6+vrd3d3q6urf39/c3NzbRDf7+/
vb29vW1tbZ2dk+D9arAAAALXRSTlMA/
elxBfACHj3YwhYJDyfOtfr2WHObgEqpjellgY6c6mZLV3L2qKjOZemN
+rUv7NpAAAACcUIEQVR4Xu2U53baQBQGAxiMaQZT3Wt6rnql9uKW9v4Pk5XEspG4
WFmJ/
GMeYM635cy7/8yePXv23N7uxnNwdHSwE1ENoLYLTz4NkM7vQHQFhKv4nlQCCIIUbF
EOXHJxPVVYUY3nKSSpKFmIJbqANRdxPIcnTHRyGEPUAga0onuy4CMb1ZM5Bh/
HmYiiGwhwE81TvIYA18VIokvY4DKKp1SBDSqlCKIyIJT5PU1AafL3FVC4q1uDLdS4+4r
CXd06bKXO21fKdxcNKFzVzUFQ9KoDJRehr0z0tBgBpcrfVyZavAxM7uo2ABHO8s2N
TW4+8pEs4HUMxS+6rYEZFBY[l]+Y8FQ3C5holImGMNVVvLp4X1FRZ2n0BU035X
+u7jmgomWv/zzWR23lB7ich/
b1DBeRQVNNty1FHYLDWVh1TwEXic4gs6105UfPdMrd1/
HCffrnqTayrYkqP3qmSom7r9q84xyMDLIUx7Mylfn7OvIGmbYyUYmGbmry99UWnYOZ7
sFc5OHb1b2HLbS9QV1vED3dPX9fQXFuWiEeF9kzpfMcfaUZkbyn9+bQTfXwvm4W0iA
3RNdQUyIV1lekkDOhSx3sdGh17+At0ctA0plDpr/gLqSvSI
+kXt8mAj9DVl20r3hGBM0iOwKmBtZXHOPX79fFXCJ/
Ure79Gxr40Owuu9hK8LPp1mnR7JmWuv7ZsM+oH3FGZObFoWx3iY/
Ul1J2LVn0b7i6B2RZM1WumwL833MoH3FMQ1hPHIGsadnvk94X3EsctPE4xK8p89FvK
84ExlRuoUa6Pt98fU1DJUMosj+3/S15OtrGOsB9LnYqm+e5w
+es4JAhYQDdgAAAABJRU5ErkJggg==) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABPl

BMVEUAAAD////////////////////////////6+vr7+/v7+/v7+/v9/f36+vr7+/v7+/
v//9/
f3/////////8/Pz////5+fn/////////6+vr/////////8/Pz7+/v7+/v////k5OTh4eHf39/
e3t7g4OD6+vrbRDf5+fnv7+/x8fH09PTY2Njc3Nz39/fq6ur4+Pjz8/Pn5+f7+/
vZ2dnW1tbt7e3U1NTV1dXl5eXr6+v19fXy8vLs7Oz29vba2trb29vo6Ojm5ubu7u7d3d38/
Pzp6enj4+Pi4uLw8PD///
+BQ30nAAAAQHRSTIMAHhYFDzEJAifiwvb98M76+PdjqtoGpZll0c46qj1icTJ1KS42A6W
Wb2WdadYK1cvCb/cJXV+KA9qioPc5l6Y36uRjySgAABXVJREFUeF7s1kuKwlAUBNDS
+JtqggkkmODIH4gPRIgD11P730BDI6HV260Z1Wu4ZxUH/
4tzzjnnnHPOOeecq2tEJdlsEsRkT
+4RkeGRPA4Rj4YkG0RjfCLJ0xixCPwWEIkd73aIwuTMu/
MEMTiwc4AeBhd2LgPolfyhhFzLBy3Eplc
+uE6hteWTLaRGCz5ZjKBU8UUFodmNL24z6KQ0pJCpaaohksxpmifQKGhjIXurSbbZhr
9qRG81yTYbaJJtNuefcsFbTbLNZnwjE7zVlNtsybdK4Vttreytdlml/
zlUvTWjmyzFT9USd7akW02pUm22TV7WOveKttswV4K1Vtlm12xp5XmrbLNBvYWhG
+15YK3qjab0STb7Fcv5tmcOBKE4Q2Xc7KrTJW3yhs/
2RsuZwvBjsQShPd0JxCyJVmA5v//getpgYextLSYKdHf
+PbUO93No36Ht77eqHCvNnvASaBLp4rooClvpYGurpx92Sx6Kw3Uv873ZbOHvA7Qtd
12m7ZZ6a00UN
+225a7F5tt8XpA7Y61rJi1ViPeSgPZ7XZnuVh4zdsseisNhAEtzgOvaZt9yGsCdSwLeJKk
16DNSm
+lgSCghQAax9NGbfaI1wQSD7YIgnE8H0a3iY4a8FYSyMKAxgJoMqBs1txbaSDs6AR5
Zv6Asllzb6WBIKAkiefzCQA5A8Jmjb2VBtoMKHXcUSM2e8x3ADoPAAh4Jr7v5G7IVKLjh
r21DHS
+DshP0xyAel3CZk29lQa6ebAUeLxe74KwWTNvpYGSFVDqOK4LPFF0QdisibfSQBsdj
QFNo9EoI2zWxFtpINiJw+Fs5hcBTaeDwaibETar76000GZAnggIeLqSyNxmD/
luQOM4LngceDBP8ACQSnTY+L111geRtqxIMfLDAsiFkV8H1GWSyNBmW7xO
+eiJyo7O87B4MCaA1Ixazd9bUwgIgFYd7a8Dwo4GIqYQgc02f291kEcd
+YJHVmZusye8djkAlKxHHnjEiEUIVOTDFKKTPdxbXdnRjghl7GjgUSqTNmvurTRRkMQ
AtBp53IkjJuNRiY72cW8NiwdLHdyJgmcdkGTKTGz2Gd
+xPOhoGVAUjZgCo8zas33cW3lviFqWhyF2ND7YbaZM2qy5t9JEs8ITPexoJlFksUzTZo
+5Tk3FiIWh7KAKpEzarLm30v/2ckczlYXdsGU6NvuY6wFdJjn+y4OXKR3EytP/
WMNbNYD6Y9Ag8WCs/F4yo51t9oBrAoGJzJV/
+WokJDow9NbaF7QJAKKkNpMyb9KPfjL2VBrLtznLpQ0DlYNTKdrHZp1wfqA0iEjhMYsisl
N9I9LSmtz7RBxLmGCSxW9k9pYye1LPZR9wACNQagIbutgaSfvTl3Fvp
+xDwgDd6ZERAVM9mT7kB0HIZJJn4Ekp7W+KRRKfG3kqfYyAg4HHy6ZZwpB
+daHgrVfG///
3z5u1l38aODsbFrcH1opJ7VPgRabNnfPeaI8+13REdPV4FBA4CRkQyZWekt2rU8M3lV
REQns9AG4XmgxLRjc0+v0d4q1ZNVjznNw8WgsTiviaRXmh5K/19jzsaL
+biwVwhsbeXdLUfffa+nrfSX9P4rYjnRcdFBWGSQ+KUf3/
ybp7vuX6lFp7woaN9eDBxiUEIwCDH/
wfCWzUrXwTrgLCj14rGuhTSxx8Q3qpZLnSQ4MlXHS2He8u
+xvqJ8FbdCmEFTXxcQdFAJkPP2qfVNvsLNy1vjg9WBKQsQMKPfiW8Vbt6SkdLFkb50
VeEt2rX1MGA4MHUlhkRfvTFh4S3aIcEHaSczmr20R+Et+rXCDu63DJsux/
9+R7hrQZE0NFEHIVh/UV4q0F1B6zSEdlWF/ISsdn/
AccLgKctjBheAAAAAEIFTkSuQmCC) 2x);

```css
}
@media (min-width: 240px) and (max-width: 420px) and (min-height: 401px),
(min-width: 421px) and (min-height: 240px) and (max-height: 560px) {
body.safe-browsing .nav-wrapper {
background: var(--google-red-600);
box-shadow: 0 -12px 24px var(--google-red-600);
}}
.safe-browsing-billing .icon { background-image: -webkit-image-set(
url(data:image/
```
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/gIDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzXXRD
XXQzbXQzbWQjXYSDvcWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP
RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLIPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////
PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTlMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnUFkRXwLiK78llgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPIElTplRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3e3DiHUhftKIlzdKoRSFe0r8sXDAkSoumkIigYaIOklf
eWi56EESFm8r1w0fFl4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRl/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXXiz0987BBtux65qaCeF8IHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7IcfH9E3ZIEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9r kJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKFtf+huwrjaa6Pxv7RfgW7nubJPtKl/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAAEIFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvcWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56lSgiaEjHwgGh
```

AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMlL
5W6t4Zkm22Ikm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjlTvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYl0gGFkvaR779KKZUWuFKe7nlIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGl8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjIMbDK01U0OVZxcplE6BIjLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRl
+oQoGTk2Z1YQyIeiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFI83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiljRDx
KQwJk9a+o2u431gYZERBo/kcBlfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzIHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPl
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
}
.safe-browsing-enhanced-protection-message { background-color: rgba(0, 0, 0, 0.2); /*
black 20% opacity */ border: 1px solid rgba(0, 0, 0, 0.3); /* black 30% opacity */
}
.safe-browsing-enhanced-protection-message .icon { background-image:
url(data:image/svg
+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdmciIHdp
ZHRoPSIyNCIgaGVpZ2h0PSIyNCIgZmlsbD0iI0ZGRiI
+PHBhdGggZD0iTTkgMjFjMCAuNTUuNDUgMSAxIDFoNGMuNTUgMCAxLS40NSAxL
TF2LTFIOXYxem0z4LTE5QzguMTQgMiA1IDUuMTQgNSA5YzAgMi4zOCAxLjE5IDQuN
DcgMyA1Ljc3YzAuLjU1LjQ1IDEgMSAxaDZjLjU1IDEgMS0uNDUgMS0xdi0yLjI2Yz

EuODEtMS4yNyAzLTMuMzYgMy01Ljc0IDAtMy44Ni0zLjE0LTctNy03em0yLjg1IDExLjFs
LS44NS42VjE2aC00di0yLjNsLS44NS0uNkM3LjggMTIuMTYgNyAxMC42MyA3IDljMC0
yLjc2IDIuMjQtNSA1LTVzNSAyLjI0IDUgNWMwIDEuNjMtLjggMy4xNi0yLjE1IDQuMXoiL
z48L3N2Zz4K);
}
</style>

<style>/* Copyright 2017 The Chromium Authors. All rights reserved.
* Use of this source code is governed by a BSD-style license that can be * found in the
LICENSE file. */
.ssl .icon {
background-image: -webkit-image-set(
url(data:image/

png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAMAAABiM0N1AAABAlBM
VEUAAADcRTfcRDfdRET/gIDcRjr/
ZmbjVTncRDfcRTfcRDfdRDzgSTncRDjeSDvcRTjbRDfbRDjeRzvcRjfbRjjcRTjcRTjcRTf
dRTfcRDjdRTjcRTjbRDjbRTjbRTjbRTfcRjjdRDrcRjfbRTjZQzfcRDjZRDfZRzbWQzzXXRD
XXQzbXQzbWQjXYSDvWQjbbRDfOQDPSQTTUQjXCPDDNPzPJPjLGPTHVQjXMPzP
RQTTWQjXLPzPDPDHYQzbAOzDTQTXHPTLIPjK8Oi++Oy/
FPTHEPTHPQDTQQDTUQTXBPDDKPjK/OzC9Oi/////
PQDPRQDS3OS66OS7TQTTEPDHXQjbMPjMBhLaWAAAAL3RSTlMA4tgPAhYFCcL9
8B4x9ie1+s49WICbqXNKZY3pjuqcgVdLZnL2qKg9zmXpjfontV8LANsAAAJrSURBVHhe
7ZTnduIwFAY3ARIgBAg9vW1v173ROylby/u/yso2Fx3MNaxs9h/zAHM+Sfa8+M/s2LFjx
+3tdjwH+/
sHWxHVAerb8KSyANnNUFkRXwLiK78IlgJHJxRalwSMd11OGOeV4nsM9FO0dxhJdw4Lr
OJ6jYy46PoohqgEHatE9JViiFNWTPIEITpIRRXcQ4C6aJ3EJAS4TkUQXsMJFFE+
+CCsU8xFEBSAoiHsaQNIQ7yuQCFe3DiHUhftKIlzdKoRSFe0r8sXDAkSoumkIigYaIOklf
eWi56EESFm8r1w0fFIl4epWgBA9qOMpmirCfeWijtoa9WSx6taAELFBRl/
vilS3BJRIbRk9/VFTsLrifUXRuNfXLU0y/7m6p0CKxqN+v6lJU/
k3eJxu7Os5LWKDHi1tYstKG1zON1X3DGiRMR80Mx3fdCbc1+bQe3o2SJrYXcV0fFMx
L9xXiz0987BBtux65qaCeF8lHCR3FabBTQ3xvk4M1yN5B/
Mw2+urew8hTP1BM38Qnu5evK8gMw+7lcfH9E3ZlEBfMSO//Kf35+Cm6ua
+rhbSYDeEa9CUyW3qK1HIjj5DBz8dWd0bWCd6Ult/uMPEr+BmbV/JHrVG/
a9MsEybV5fsK50R3frmBFXtCtVXmt73H4PhQ4t9k9rkJ55tYXwZrO4rCEUfPHfUEcuaZC
/umw97TfaVpslu2tCb2lRWnBlKFtf+huwrjaa6Pxv7RfgW7nubJPtKI/X0puQO4k/Pfe/
ovtLY7KbxVwve0/sE3VeaLosIbkEDvt8Hoq/hKGwQYvoq5OMnoq/hLAbgc/
FVn33PX7pAfE5QHR6fAAAAAEIFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzbXWQzbXQjbWQzXZRDbbRDnWQjXWQzbXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvcWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////

bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56lSgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xzzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMlL
5W6t4Zkm22lkm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjlTvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYI0gGFkvaR779KKZUWuFKe7nlIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW
Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGl8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjIMbDK01U0OVZxcplE6BIjLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyleiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFl83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiIjRDx
KQwJk9a+o2u431gYZERBo/kcBlfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzlHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPI
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==) 2x);
}
.ssl .legacy-tls.icon { background-image: -webkit-image-set(
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAEgAAABICAYAAABV7bNHAAAAAXN
SR0IArs4c6QAAB21JREFUeAHtXF1IHFcU9ie2bovECqwWxeWyLjRH60BYpKZHYpoFC
U60/
xKCt5ME3QaSpT6WUPElCEXyTUpIojfgTUwshNpBgqZVQ86hGktdgSsFGQqr1t9+nd2
WZPefO7LjzjYzcJmZc8495zvf3Ll3zu

+dzcoKt5CBkIGQgZCBkIFMZSB7r4G3tLS8sLCw8D7ivo1Ssrm5WYL9AZSC7OzsAuyzI
HuCHcsjyOawZ7lbVFT0W09Pzz843rNtTwhqaGh4ZXV1tQFZfYZSDgKe85MhyFpBvTso
V/Py8q5g
+9OPn0TqpJSgurq6CpBxFuUEQO1LBJgH2zUQdgPlwuDg4LgHe18mKSGovr7+2Pr6+j
kgOuILVeKVJnJzc78eGBi4nXhVe42kEtTY2Fi8vLz8HVrMKXvY1GjRmvrz8/Pb+/
r65pMVlWkEodV8vLGx8SPI2Z8scH78gKTFnJyc02hN1/3Ud9ZJCkG1tbVfwnEnyMlxBp
DOkcQybG9ifwv6OezvRyKRv5eWljhyZeG4AMcvweYNnHKkq4TNcezzqXfbYLsBm46h
oaELbrZu+l0R1Nra+vz8/HwPgH/uFgj6xwA
+inINt8Evvb29Tz3U2TFpamp6EbfvR4hVhXISisIdpXKAWJeLi4tburu7/1VMXMW
+CcII9TKA/oTyni0KQC5B34V9J0abRZutVx1i70fcDti3YR
+x1UPcSZRPEfsvm52m80WQaTm3beQA1Dr0F9EffANwDzUAu5GDqIPo975FrGbEytV
8QT
+JlnTMT0vyRRD6nEsAZLutOIpUDw8P86Eu5VtNTU05goygFGvBQNJl9ElfaHpNrrKuV
WCHDHLOanoAmUKr
+QBgZjWbZMtnZ2cflpWV9cPvUZRXf9vHT58+OnMzMzvil4UJ0QQh3KQ8wM8iS0P5P
SjVOGWWhCjpVCIxJ+AgD6EeA2lTAoFbB
+CyKnp6en7kl6SiYlKhuYhcBYEic85JAethu9bad/Qyq8Ap/
iwCpyLGEUPeX2Y9PTcwozNE7JGzhQCn0k7MwYAsaBMSXh4gZmLpJNknlqQebe6JT
mAbB59zru7GanQyW5KvtHJe8In1TUj3B/
QiR033t0qvby7eWpB5sUzDgeu0jqE1bshJ85pkgQGU7XBGOdVy8lp6EoQrkQFKolv5Wi
uF/dqKHcC93JObMSo2B4xuSnqbbErQQQggDum4Mkt8CLR6D4CSGIlVgqLlFmtrJYi/
BMIJf+yStq4g3lpOoAZjl1POc+bGHCVdVGYlaGVl5TQMpV8C+eLZGXUS9L3B+ljAuc/
8FCyotkVS8jvGcFwNlnfOoweQj
+LKJOXFkz53M1pFMdn2xlpno1HkIr0e8XdysYXRp9qCOPsAPd9x4jYQdC1OGHCBBX
O5yVXMQCWIUzNgPG72AYGW
+XuO6C3AQmImdidE5mimoZyqrXOVIGg5bxW3weHNRH/
sinOSBgExE7sSWsyVtjaCSiRnuAraE7VkHiiZBbuYK8GrBIFtsRKC3AtU1gmA0bBrudK1
bRQ7oMR
+oMh9i1PxLqaA0bBrueotCAG25smdgTj74JRlyrkFu5gr81JvMTRHsVJ0aiZTSInFqWHX
crUSFOv4WT5WWxA6rq1JPCc5nNRzyjLlXMOu5cq8VIKgEwnijGemEOLEacEu5sr6NoI
eOQPwHGxzOjgjNwt2MVcmqRKEjmtOYUF8PlJsgyYWsVty1QlCZiJBuAqVQcvaKx4Ldj
FX+lVbEHR3pcBg
+zgXEki6IMulmdgVjGKutFUJ4oJJOFxxOsRVyOcqC6c86OdmZUjc8hnmyFw1/
CpBZjWpOLcOkqo0h0GVWzDfsa2cVQkyiV6VEkawk5gRECcRJft0y4iVmBUcYo5RWyt
BXGoLw7Woccy
+EAE7Ys4DfWiwFgog10yOgmpbZCWI65Bxj44ptdtwZQ4qusCIDcY2CRByu
+G21tpKEJ3CyXnJOa5KhIuXJF2QZMRIrBIm5Oa6htGVIMwIjMP5hBKg2SxektRplxEbS
GhWgEyY3BT1ttiVIJpxkbbkBVeG64tGgnirGUwjBmMcfC0np6Hn1RMua264/
OUorog4xesMmupzkBMBMb
+ivCPFAlbPa5k8tSAGwbRJOxyLk4UEgsKVZ4HYiMVCDhdQtXsF6rkF0aFZTf8zgovE8s
qgnEIXSzlth+SckggAtg0sZvgkkVX4Ca1R5Nq+0tJSfq
+lvWpwbeAJrBW8zjWDEshUydjngJgxFA0bR+SvcPEuJYIhoRYUdYz
+6JlZBizeKlEitD2X9+NqTGp6yIuhn8Aw+70ZTSym/lX0zRiMxZiaJ2llZk1vk/
tqQXQIcOGnCDZmqQs/ZnFjyOjRJ/n+HArNn1PZDzipF5234uyD
+YH9dXS6b6Jk5udQsfz9Xz
+o89VJxxITPeazBR7ADqFF8JuJtGyMTQyJPOe4AfXdSdscm4Xn52AjLh

+21fWpy4yPep3JYaSrQP+Rys/
Cx9BqzuPhb9wZO1nnKWlBTnDhHws4GbGcZ9pfU1hSCVUhAyEDIQMhAyEDAWfgP5
qNU5RLQmxEAAAAAElFTkSuQmCC) 1x,
url(data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAYAAADnRuK4AAAAAXN
SR0IArs4c6QAAEp1JREFUeAHtnVuMFkUWx2dgRlBhvUxQSZTsw25wAUPiNQTRgFkv
8YIbZhBcB8hK2NVkXnxRY0xMDFFffJkHsyxskBFRGIJ4iWjioLJqdL3EENFZ35AELxnR
HZFFBtjff+gePsbv0qe6+vv6+6Y66XR39aIT5/
zPv6urq6q7m5rCEhAICAQEAgIBgYBAICAQEAgIBgYBAQCAgEBAICAQEA
glBgYBAQCAgEBBoTASaG9Ot8l6tWLFFi4sGDB3+P1HSStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTTpyeeeeaaZX65fv/5/HI
+pZUwQ6l477vjD0NDQAgiwgOBfynYYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGGGGF1dXRP2799sCPLMhuGGGGF1
glBgYBAQCAgEBBoTASaG9Ot8l6tWLFFi4sGDB3+P1HSStx44d0/
a85ubmyWwnHz9+fHgbHTdxPEj6IMfD2+j423HjxvWTTpyeeeeaaZX65fv/5/HI
+pZUwQ6l477vjD0NDQAgiwgOBfynYYkUEdHxxSCuBDAF7DOZ/
+CWoAPmb6m3J2sfexv37Jly3e1sCPLMhuGGQF1dXRP2799sCPLMhuGGQF1dXRP2799/
G2TpBLCbWFuyBM5B9xB5XoVlPVOnTn2xu7v7slOO3GWpewJR21xJG
+ZukF3MenbuEC5u0A8kb6YNtY5a6YPiIvWRWrcEWrx48XyI8xA1znX1AXVxK6mR3oBl
qzdv3qxbXd0tdUcgapyblY2IM6fu0C5jMER6j3U1NdIrZcRyd6puCARx5kCabtbLcoeiR4
Mg0UesXRDpPY9qM1OVewltW7asjT6bJ0DgL6y5t9dTpI6j55/0Ld2/
YcOGAU86M1GT24BQ0zS3t7evxOvHWNsy8T7/
SkWeB3t7e9dSK4lUuVtySSBuV9NoID8LWnNzh1htDHqHhvad3Nb21qb40qV67Y0tXUz
yMzxd3Urt8wk5AnlOwjZXmAibk0n52MtNDbRq1arWgYGBx4HlvmpAwy3hJ8rpJzD98Zg
W+1+RPjh+/PjB0047bfDQoUMa+2o6/fTTJ//yyy+Tjx49OjxOhsxFA+PobE/
PJ5G3kmSrcLyZFbtb2wNr1qw5UoWyKhaRCwltWbLkIsqthCEqypa7CggwqD/
bbZ9bPsuueSSTx955JFFjjupOyYaecbt3756Nbo21acztGraZEQr97zPW1vHcc899dYohN
TioOYFo78ygvfMavMavl+Ygf8aQe
+lhumZMWPGLgKt4YTMF8pp62bNnzzz86oRI7RSo0X3fiyz78uoF20R7fii36akqgqG/
nZUA+12J0JVll8zrr08htA+BDIeSzPM+t+YwDBw7cjo/LWa/3WRY
+fs96Sy37jGpGhMzM1foZgA9wweoAKnb0VbaL6uZRvGpD52+dTCtZDbtqlfQuwgy
+XqA+ZmaaDEkqkkkPdk0IRP/
OnwFwPUCmHjGPiPNMa2vrY5s2bfrCd9Cz0Ld06dKLjxw58iC67/
JEpCFItBwSqeujqkvVCRTVPC/gpQ/
yfEgA7tm6deuHVUXNU2GLFi26nAvgKXy43INKkkej2atdEvqrRRP6rzRPdtlKRB9APANa
9s2bNuqpeySPAZLt8kC/yKRGIpYVahK0wLi3i/
0zVaiAcm8GVtos1VYMZoHfQL7O8p6fnW/
9w1E5jZ2fnefQ7PQ0+N6axnzz3wWYYNl/
qqqlfZt7MEwCUy3GfhNNIfXNzz3wWYY1/
dQkj3KYppbeN3Hixd+8803tdSfl9TzefPmTUyWkxpUdghDn7MXr9MQIwlwSUJFIMoFpn
+S4MhEUpnCqlKwaKRSU51/
OZEIgrphnDn2Xr9MQIwFg7xuKbnqMDKKyEhSuJFIMoFpncbTlhUDST0Gk
+D0C9xVWnyVNHR4M5Vo
+FhTARNo4YzI1i4pi9dDbvrIzmMPdTpMs0VDWYrx3Lt63SoWpqUpuI2kQkml1OrsS5Ae
ZYT/c9x9p7DRRNgHchjx7Vx3Sbp0TgR5J1YQkjElwe8eOXE0b0+djxWgNxhWio4h0Ms
+pVJ6H6eWr2qM64lKlzkmEq48+4jWsA5yvBuedHLQYlR4H57ng7O2Vla81EA22bhwy

A4tTD9eSPMYg1FxcWAkzB0Oaoxg5ZC2exRuBuCr0xuhlxYspnUrDcleGJ0pLhDPFEli
GdHYUO1cuTTFSrMrJWM55IxCGaaKUaYE8BzQwytZ0+zAV0qDCwizCzjyK7xKrUjB6I
RA9zvoGj3kaASA81Gij6qWAziJd2AlDq27FSjGz5ism74VANOjMTuD4hzNnzvx7MaNC
WnlEhKGwTJ7jhKRLzIqVkZpA3E
+vhNGmT6zgsD4Hd4+v12qKOTZW0oShsBSmFp8VM8XOkqeYbGoCYcjKYoorpD1Tzz
MJK/hW9dMRls9YC3aM3SnFpCKQPiuHER2naKxwoCtFE
+AriIXTRgSEqUMt1KEYGos6RTwVgfRNQrRZPyu3tV7enjgFqZwfRJhuNZp5dhRDY7aT
4qkIhJplJ1Ul29N7W8kkg5QVARdsuYPoo6TOizOBaIDpU7qmCeBUsa/n9aU/
ZwRzlFHYCmOjSTcplsY8I+LWsZSRjJBnIQem/
Dj39IiCnO3UcmzLJxTCmNhYXqFuiWK51sUO5xqIwhYYCxxE3nlmnbGssSwujIW1ZbH
GckR3GgKZejK5MnoZBKzphw5GvG7gHWEsrI0ummJZqNuJQNwz9ZKg6fcBjB73FBYc
9rNDwIq1Yqn/ibhY5EQgusFNjOWK
+Enf53ExMOSxIyCshbklp35GY5GPZZ0IhHGmwmD429X6uFPs2FjeCmthbsHAGtNYtx
OBMO7SWEGSLcb1JZELMv4QsGJujWlsqZlA
+lkbxpneM8K4QKAY8SptrZgrpoqt1TwzgfSnP4xLnA/DftIHLa2GBfl0CAhzYZ9Ui2Ia/
cUxaZZhucREKNCqz9palv4wbcMClx/
ZCHO9XmVZrLFtypxAMNvqhMXhIFsGAQfssycQj/CmQuiTCAQqE
+QsT1mxt8ZWtpvGspSB+
+r5MFu7SZe6IFA9vReWFHjkTNgrtgbdw6IutzDTR7Mh21dWo4K8HwQcsDfFVla6EMj0
CX9YbR3Y84Ne0KK7hRV7U2ydCASrTSxlkpPViRB6TwhYsbfG1olAZDIRSH
+98YRHUGNEwAF7U2xljvkWRrVoKiT
+ZZLR9yDuAQEr9tbYykQzgTz4FVQ0EAJmAnGfNN2S9LO2BsKrrIyxYm
+NrcAwE4g8JgLpT391hXoDGeuAvSm2gspMlOujoX4T2UAxqStXrNhbY
+tEIDKZWOryaFhXUcqxsQ7Ym2LrSqDEUwRUAKzWD2rDUgMErNhXpQ1EId8YsTAN
vhp1B/HyCFixN/
8BydwGqsYIb3lMwtmkCFhH162xlR1mApHHOsJrvQqS4hPkKiDALcyKvSm2Kj5zAlHGd
GbHuZRTAZ5wuhwCEeb5IxBfO/
8SZh8rZ3zhOdpMk3bv3j27MC3sZ4+AMBf2SUtSTBXbpPKxnLlm0M8/
MGxvrCDJFuMWJJELMv4QsGKumLr83MZMlLmIcR9bXMW4QCALYB5krZhbYxqb6EQ
gjDO954Vx13BPNk
+fjY0MWxsCwlqYW3JZYxrrdiJQS0uLiUAYN2nPnj3z4kLDNIsEhLUwt5RijWms24lAfAnrc
xj+dawkyZY+iVSfUktSRpA5gYAVa8VSMXXBz4lAUUH6W0zihSuinc/
CnJ44QxB0QkAYC2tjZlMsC3WnIZDpNkahGpX/
U2HhYT8TBISxdQaENZYjhjsTiGpvO1qGRjQl2OHKWJ5ALIikQACMVxizD0WxNGY7Ie
5MID6l9h0qXrWUinPX8yWs0KloAc0gK2zB+I+GLBJ9NYqIMdsJcWcCKTvMNX+2jklO5h
+zOHk2BjO5YOsSw0JoUxFo6tSpL6Lsh0KFCfYXLV269OIEckHEgECE6SJDFon
+EMXQmO2keCoCdXd3H0bV5pPqKu9RxY47cuTIg5Ulg4QFAWEqbC15kN0cxdCY7aS
4tcCTOaM95pCs+1Vi5YS7+JjB5ZXFgkQSBCIs70oiWyjjGLtCFU7TOU5RQAPsA
+6jb5ySWOFAVwp5ngrTPCoAleC0MBSW1tpHMVPsEhRRViR1DSTtMNn8AxUcvvyzzz
77a1nrwsmKCAhDYVlRcJSAS8xGqRg+9Elg/iC8E0a/V6yAcmk4vrqzs/
O8cjLhXGkEhJ0wLC1R/IxipZgVP2tL9UIgFYlRZkdw/
hze39bPQZptZgdpYRZhd44VDZdYlSrDG4G4n76CYR+VKqhUOkDcyB
+E7y91PqQXR0CYCbviZ0unKkaKVWkJ2xIvBFKxGNfF5rjNhKYmRo8fZRDwamu
+sSovrlSZg//Hoxg5ZC2exfutg0fKtRR1d/Hiyqbuo2F3BVeHaZpIWY0NeBLyXAB5/
o1rFzq4t47/oq10yFcyi9caSKUwMVu3o4GSJZY
+cSHA7ACgs0qLjO0zwkYYgYILeQai2HgF0TuBNmzYIPK49jRrMHC7yyf3vaKSQ2XCR

Nhgmutg9INRbLx65/0WJutwtLm9vX0Xu3NdrOU+vY21g9vZUZf8jZaHmmc8mG5h1Vwfl
+Wd3t7eeWBqbp9WKsx7DaQCZSjtmTvZfl/
JgGLnBZQACzVRU1NU8ziTRzGIYuGdPMOxLhZAX2k8at7KFAON2DstOP8W60Jqoh
+dFNR5JrV5uJC2s17r6gpfar2NTsOXXPNXyje+kkCa83Sz/4e/
5/0GHXMc9fwW8G6aNWvWC7xpYPqsjGN5uckGefS0pTHGq1IY9SS3ru4U
+StmzeQWVlhqW1vbA9Qi7xemGfdn67EVQMdMP5F8lc/
g5NpgVjPifWFvxNosnkkjerQVS5YsuYj5Ku
+S7vL4Gasb4l7+MNXxE4CTyf08LqhWW2rbZvUwQx51EqZ5EXPfxIkT52zcuHFf1r5Uh
UBygqtKf3rexXpuGqcgzw6+Prq8p6fH/
DGkNOVmnVcDo9HYlnl4otA28PmedR7txj2F6VntZ9oGKjSaNsx3M2fOFlGWkt5aeM64/
zv+MLwSXf/
lav34zTffrOvaSPN5pkyZ8jdq6G1gc4kRi9HiP1NL3wh5Phl9IqvjqtVAsQPURDdTRb/
AcZoqOlandsK9dM9/GCfU01YzCaktNBnMPJ+niJ
+6xd8OebwNlBYp41dJVSeQLIBEd0Kip9lNTSICcAw9z7S2tj62adOmL6Q/74smwEfzwu
+CPD4eZESe5ZDn2Wr7XhMCycmoJtKE/DN8OB0RaSv9Hqt5z/
tTHzp969B7W9GrN4s8EUcm6ra1uNo1T4xNzQgkAyDRHIB8mTVVwzp2Jt5CptdZVcNt
A9hDcXottvio7wGoZ3056/U+bcBHNZhvwUfzbFBfdtSUQHICgGdwO3uN3TSP
+KXwGATgXq7QHjo0d9FgHSol6DOdclr0iRX86oQ07eie7FN/
pEvTX26APFV52iplf80JJMPUT8STlcZ70vS6lvJxOB0i/YT
+t9n2se3Tf9UJtNpPqRc9SembhOhegO4FbK9ha/o+j8UI9L8/
YcKE9mr081SyKxcEkpGrVq1qHRgYeJzd+yoZ7eM8QdDQSD+B7udK7o/2vyJ9UH/608/
a4v9t6a83+nEJ7ZfJyE9G5iLkp1PDTGdfX0KdniVh0F
+4PKke5jVr1hwpTKzVfm4IFAOgAVgCs56AeG0XxfrrdQtRNaq
+IsuBURdsckcgOUG7aBok0iOp03wiFyBynucdyHMn7Z29ebMzlwQSSNRAmpS2kt3HW
NuUNgaX4dmdjKivpQbKZY+7j06sTOIqwOhh/gfzeNXGWMeaSwAzcf6Er
+vkuzDIK3nke25roNGBifqMuqmZLht9rpGOIctHrF217Nux4Fk3BIqdgkg3Q6KHWF0nqc
WqcrWFNO+xroY4VR3LSgtC3REodpintfk0tEWk6+K0etxCmjdoIK/
29a56tTGoWwLFQFEjXQmJVrJ2kHZ2nJ7z7Q8QZwvrWmqc1J9YqaWvdU+gGLyurq4J
+/fvv43jZZBJk7JSj/THuj1t9TVUvRS4QZ+VS/
tIME82pVbTMAQqRIJaaQokWkjaAtb57F9QeL5a
+xBGr2nvZO1jfzu1jb5s21BLQxJodlQglAZs5xNEjVVdynYaW69dGOg8hs69bD9m20e7Z
ieEqelA52gcsjgeEwQaDZxe1jt48ODvSR8ex4JcGtM6n2ONmk
+CANpqzGt4FJ3jQY41sq+txtAGSfsGkgyPoXHcT5/
Nly7/2yJvWAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIBAQCAgEBAICAYGAQEAgIB
AQCAgEBAICAYEcIvB/Q079+h6myXwAAAAASUVORK5CYII=) 2x);
}
@media (prefers-color-scheme: dark) { .ssl .legacy-tls.icon {
filter: invert(1); }
}
.ssl-opt-in .checkbox::after { border-color: #696969;
}
.ssl-opt-in .checkbox::before { border-color: #696969;
}
.ssl-enhanced-protection-message {
background-color: rgb(248, 249, 250); /* google-gray-50 */ border: 1px solid rgb(218, 220, 224); /* google-gray-300 */

}
@media (prefers-color-scheme: dark) { .ssl-enhanced-protection-message {
background-color: rgb(32, 33, 36); /* google-gray-900 */
border: 1px solid rgb(95, 99, 104); /* google-gray-700 */ }
}
.ssl-enhanced-protection-message .icon { background-image: url(data:image/svg

+xml;base64,PHN2ZyB4bWxucz0iaHR0cDovL3d3dy53My5vcmcvMjAwMC9zdmciIHdp
ZHRoPSIyNCIgaGVpZ2h0PSIyNCIgZmlsbD0iIzY5Njk2OSI
+PHBhdGggZD0iTTkgMjFjMCAuNTUuNDUgMSAxIDFoNGMuNTUgMCAxLS40NSAxL
TF2LTFIOXYxem0zLTE5QzguMTQgMiA1IDUuMTQgNSA5YzAgMi4zOCAxLjE5IDQuN
DcgMyA1Ljc0VjE3YzAgLjU1LjQ1IDEgMSAxaDYgMS0uNDUgMS0xdi0yLjI2Yz
EuODEtMS4yNyAzLTMuMzYgMy01Ljc0IDAtMy44Ni0zLjE0LTctNy03em0yLjg1IDExLjFs
LS44NS42VjE2aC00di0yLjNsLS44NS0uNkM3LjggMTIuMTYgNyAxMC42MyA3IDljMC0
yLjc2IDIuMjQtNSA1LTVzNSAyLjI0IDUgNWMwIDEuNjMtLjggMy4xNi0yLjE1IDQuMXoiL
z48L3N2Zz4K);
}
.ssl-enhanced-protection-message a:link { color: rgb(26, 115, 232); /* google-blue-600 */
}
@media (prefers-color-scheme: dark) { .ssl-enhanced-protection-message a:link {
color: rgb(138, 180, 248); /* google-blue-300 */ }
} </style>

```
/**
* @fileoverview Assertion support. */
/**
*   Verify |condition| is truthy and return |condition| if so.

*   @template T

*   @param {T} condition A condition to check for truthiness. Note that this

*   may be used to test whether a value is defined or not, and we don't
    want

*   to force a cast to Boolean.

*   @param {string=} opt_message A message to show on failure.

*   @return {T} A non-null |condition|.

*   @closurePrimitive {asserts.truthy}

*   @suppress {reportUnknownTypes} because T is not sufficiently
    constrained. */
```

```
/* #export */ function assert(condition, opt_message) { if (!condition) {
let message = 'Assertion failed'; if (opt_message) {
message = message + ': ' + opt_message; }
const error = new Error(message); const global = function() {
const thisOrSelf = this || self;
/** @type {boolean} */ thisOrSelf.traceAssertionsForTesting; return thisOrSelf;
}();
if (global.traceAssertionsForTesting) {
console.warn(error.stack); }
throw error; }
return condition; }
/**
```

```
* * * * * * * * * *
```
Call this from places in the code that should never be reached.
For example, handling all the values of enum with a switch() like this:
function getValueFromEnum(enum) { switch (enum) {
case ENUM_FIRST_OF_TWO: return first
case ENUM_LAST_OF_TWO: return last;
*}
-      * assertNotReached();

-      * return document; *}
  *

* This code should only be hit in the case of serious programmer error or
* * * * */
unexpected input.
@param {string=} opt_message A message to show when this is hit. @closurePrimitive
{asserts.fail}
/* #export */ function assertNotReached(opt_message) { assert(false, opt_message ||
'Unreachable code hit');
}
/**
* @param {*} value The value to check.
* @param {function(new: T, ...)} type A user-defined constructor.
* @param {string=} opt_message A message to show when this is hit. * @return {T}
* @template T
*/
/* #export */ function assertInstanceof(value, type, opt_message) {
// We don't use assert immediately here so that we avoid constructing an error //
message if we don't have to.
if (!(value instanceof type)) {
assertNotReached(
opt_message ||
'Value ' + value + ' is not a[n] ' + (type.name || typeof type));
```

}

return value; }

/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.'); </script>
<script>// Copyright (c) 2012 The Chromium Authors. All rights reserved. // Use of this
source code is governed by a BSD-style license that can be // found in the LICENSE
file.

// #import {assertInstanceof} from './assert.m.js'; // #import {dispatchSimpleEvent} from
'./cr.m.js';

/**
* Alias for document.getElementById. Found elements must be HTMLElements. *
@param {string} id The ID of the element to find.
* @return {HTMLElement} The found element or null if not found.
*/

/* #export */ function $(id) {
// Disable getElementById restriction here, since we are instructing other // places to re-
use the $() that is defined here.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, HTMLElement) : null; }
// TODO(devlin): This should return SVGElement, but closure compiler is missing //
those externs.

/**
* Alias for document.getElementById. Found elements must be SVGElements. *
@param {string} id The ID of the element to find.
* @return {Element} The found element or null if not found.
*/

/* #export */ function getSVGElement(id) {
// Disable getElementById restriction here, since it is not suitable for SVG // elements.
// eslint-disable-next-line no-restricted-properties
const el = document.getElementById(id);
return el ? assertInstanceof(el, Element) : null;
}

/**
* @return {?Element} The currently focused element (including elements that are *
behind a shadow root), or null if nothing is focused.
*/

/* #export */ function getDeepActiveElement() {
let a = document.activeElement;
while (a && a.shadowRoot && a.shadowRoot.activeElement) {
a = a.shadowRoot.activeElement; }
return a; }

//

/**
* @param {Node} el A node to search for ancestors with |className|.
* @param {string} className A class to search for.
* @return {Element} A node with class of |className| or null if none is found. */

```
/* #export */ function findAncestorByClass(el, className) { return /** @type {Element}
*/ (findAncestor(el, function(el) {
return el.classList && el.classList.contains(className);
})); }
/**
  •            *  Return the first ancestor for which the {@code predicate} returns true.

  •            *  @param {Node} node The node to check.

  •            *  @param {function(Node):boolean} predicate The function that tests the

  •            *  nodes.

  •            *  @param {boolean=} includeShadowHosts

  •            *  @return {Node} The found ancestor or null if not found. */
       /* #export */ function findAncestor(node, predicate, includeShadowHosts) { while
       (node !== null) {
       if (predicate(node)) { break;
       }
       node = includeShadowHosts && node instanceof ShadowRoot ? node.host :

}
return node; }
node.parentNode;
/**
* Disables text selection and dragging, with optional callbacks to specify
* overrides.
* @param {function(Event):boolean=} opt_allowSelectStart Unless this function * is
defined and returns true, the onselectionstart event will be
* suppressed.
* @param {function(Event):boolean=} opt_allowDragStart Unless this function * is
defined and returns true, the ondragstart event will be suppressed.
*/
/* #export */ function disableTextSelectAndDrag( opt_allowSelectStart,
opt_allowDragStart) {
// Disable text selection. document.onselectstart = function(e) {
if (!(opt_allowSelectStart && opt_allowSelectStart.call(this, e))) { e.preventDefault();
} };
// Disable dragging. document.ondragstart = function(e) {
if (!(opt_allowDragStart && opt_allowDragStart.call(this, e))) { e.preventDefault();
} };
}
/**
* Check the directionality of the page.
* @return {boolean} True if Chrome is running an RTL UI. */
```

```
/* #export */ function isRTL() {
return document.documentElement.dir === 'rtl';
}
/**
* Get an element that's known to exist by its ID. We use this instead of just * calling
getElementById and not checking the result because this lets us
* satisfy the JSCompiler type system.
* @param {string} id The identifier name.
* @return {!HTMLElement} the Element.
*/
/* #export */ function getRequiredElement(id) { return assertInstanceof(
$(id), HTMLElement, 'Missing required element: ' + id); }
/**
```

- * Query an element that's known to exist by a selector. We use this instead of

- * just calling querySelector and not checking the result because this lets us

- * satisfy the JSCompiler type system.

- * @param {string} selectors CSS selectors to query the element.

- * @param {(!Document|!DocumentFragment|!Element)=} opt_context An optional

- * context object for querySelector.

- * @return {!HTMLElement} the Element. */
  ```
  /* #export */ function queryRequiredElement(selectors, opt_context) { const element
  = (opt_context || document).querySelector(selectors); return assertInstanceof(
  element, HTMLElement, 'Missing required element: ' + selectors); }
  /**
  ```

- * Creates a new URL which is the old URL with a GET param of key=value.

- * @param {string} url The base URL. There is not sanity checking on the URL so

- * it must be passed in a proper format.

- * @param {string} key The key of the param.

- * @param {string} value The value of the param.

- * @return {string} The new URL. */
/* #export */ function appendParam(url, key, value) {

```
const param = encodeURIComponent(key) + '=' + encodeURIComponent(value);
if (url.indexOf('?') === -1) { return url + '?' + param;
}
return url + '&' + param; }
/**
* Creates an element of a specified type with a specified class name. * @param {string}
type The node type.
* @param {string} className The class name to use.
* @return {Element} The created element.
*/
/* #export */ function createElementWithClassName(type, className) { const elm =
document.createElement(type);
elm.className = className;
return elm;
}
/**
* transitionend does not always fire (e.g. when animation is aborted
* or when no paint happens during the animation). This function sets up
* a timer and emulate the event if it is not fired when the timer expires.
* @param {!HTMLElement} el The element to watch for transitionend.
* @param {number=} opt_timeOut The maximum wait time in milliseconds for the *
transitionend to happen. If not specified, it is fetched from |el|
* using the transitionDuration style value.
*/
/* #export */ function ensureTransitionEndEvent(el, opt_timeOut) { if (opt_timeOut ===
undefined) {
const style = getComputedStyle(el);
opt_timeOut = parseFloat(style.transitionDuration) * 1000;
// Give an additional 50ms buffer for the animation to complete.
opt_timeOut += 50; }
let fired = false; el.addEventListener('transitionend', function f(e) {
el.removeEventListener('transitionend', f);
fired = true; });
window.setTimeout(function() { if (!fired) {
cr.dispatchSimpleEvent(el, 'transitionend', true);
}
}, opt_timeOut);
}
/**
* Alias for document.scrollTop getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The Y document scroll offset.
```

```
*/
/* #export */ function scrollTopForDocument(doc) {
return doc.documentElement.scrollTop || doc.body.scrollTop;
}
/**
* Alias for document.scrollTop setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target Y scroll offset.
*/
/* #export */ function setScrollTopForDocument(doc, value)
{ doc.documentElement.scrollTop = doc.body.scrollTop = value;
}
/**
* Alias for document.scrollLeft getter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @return {number} The X document scroll offset.
*/
/* #export */ function scrollLeftForDocument(doc) {
return doc.documentElement.scrollLeft || doc.body.scrollLeft;
}
/**
* Alias for document.scrollLeft setter.
* @param {!HTMLDocument} doc The document node where information will be *
queried from.
* @param {number} value The target X scroll offset.
*/
/* #export */ function setScrollLeftForDocument(doc, value)
{ doc.documentElement.scrollLeft = doc.body.scrollLeft = value;
}
/**
* Replaces '&', '<', '>', '"', and ''' characters with their HTML encoding.
* @param {string} original The original string.
* @return {string} The string with all the characters mentioned above replaced. */
/* #export */ function HTMLEscape(original) { return original.replace(/&/g, '&amp;')
.replace(/</g, '&lt;') .replace(/>/g, '&gt;') .replace(/"/g, '&quot;') .replace(/'/g, '&#39;');
}
/**
* Shortens the provided string (if necessary) to a string of length at most
* |maxLength|.
* @param {string} original The original string.
* @param {number} maxLength The maximum length allowed for the string. * @return
{string} The original string if its length does not exceed
* |maxLength|. Otherwise the first |maxLength| - 1 characters with '...'
* appended.
```

```javascript
*/
/* #export */ function elide(original, maxLength) { if (original.length <= maxLength) {
return original; }
return original.substring(0, maxLength - 1) + '\u2026'; }
/**
* Quote a string so it can be used in a regular expression. * @param {string} str The
source string.
* @return {string} The escaped string.
*/
/* #export */ function quoteString(str) {
return str.replace(/([\\\.\+\*\?\[\^\]\$\(\)\{\}\=\!\<\>\|\:])/g, '\\$1');
}
/**
* Calls |callback| and stops listening the first time any event in |eventNames|
* is triggered on |target|.
* @param {!EventTarget} target
* @param {!Array<string>|string} eventNames Array or space-delimited string of * event
names to listen to (e.g. 'click mousedown').
* @param {function(!Event)} callback Called at most once. The
* optional return value is passed on by the listener.
*/
/* #export */ function listenOnce(target, eventNames, callback) {
if (!Array.isArray(eventNames)) { eventNames = eventNames.split(/ +/);
}
const removeAllAndCallCallback = function(event)
{ eventNames.forEach(function(eventName) {
target.removeEventListener(eventName, removeAllAndCallCallback, false); });
return callback(event); };
eventNames.forEach(function(eventName) { target.addEventListener(eventName,
removeAllAndCallCallback, false);
}); }
/**
* @param {!Event} e
* @return {boolean} Whether a modifier key was down when processing |e|. */
/* #export */ function hasKeyModifiers(e) {
return !!(e.altKey || e.ctrlKey || e.metaKey || e.shiftKey);
}
/**
* @param {!Element} el
* @return {boolean} Whether the element is interactive via text input. */
/* #export */ function isTextInputElement(el) {
return el.tagName === 'INPUT' || el.tagName === 'TEXTAREA';
}
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.'); </script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
```

LICENSE file.
const CAPTIVEPORTAL_CMD_OPEN_LOGIN_PAGE = 'openLoginPage'; </script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
function setupSSLDebuggingInfo() {
if (loadTimeData.getString('type') !== 'SSL') {
return; }
// The titles are not internationalized because this is debugging information
// for bug reports, help center posts, etc.
appendDebuggingField('Subject', loadTimeData.getString('subject'));
appendDebuggingField('Issuer', loadTimeData.getString('issuer'));
appendDebuggingField('Expires on', loadTimeData.getString('expirationDate'));
appendDebuggingField('Current date', loadTimeData.getString('currentDate'));
appendDebuggingField('PEM encoded chain', loadTimeData.getString('pem'),
true);
const ctInfo = loadTimeData.getString('ct');
if (ctInfo) {
appendDebuggingField('Certificate Transparency', ctInfo);
}
$('error-code').addEventListener('click', toggleDebuggingInfo); $('error-
code').setAttribute('role', 'button'); $('error-code').setAttribute('aria-expanded', false);
} </script>
<script>// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h. const SB_BOX_CHECKED =
'boxchecked';
const SB_DISPLAY_CHECK_BOX = 'displaycheckbox';
// This sets up the Extended Safe Browsing Reporting opt-in, either for
// reporting malware or invalid certificate chains. Does nothing if the
// interstitial type is not SAFEBROWSING or SSL or CAPTIVE_PORTAL. function
setupExtendedReportingCheckbox() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' &&
interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!loadTimeData.getBoolean(SB_DISPLAY_CHECK_BOX)) { return;
}
if ($('privacy-link')) {
$('privacy-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY);
return false;
});
$('privacy-link').addEventListener('mousedown', function() {

```
return false; });
}
$('opt-in-checkbox').checked = loadTimeData.getBoolean(SB_BOX_CHECKED); $
('extended-reporting-opt-in').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-opt-in';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-opt-in'; }
$('extended-reporting-opt-in').classList.add(className);
$('body').classList.add('extended-reporting-has-checkbox');
if ($('whitepaper-link')) { $('whitepaper-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER); });
}
$('opt-in-checkbox').addEventListener('click', function() { sendCommand($('opt-in-
checkbox').checked ?
SecurityInterstitialCommandId.CMD_DO_REPORT :
SecurityInterstitialCommandId.CMD_DONT_REPORT);
}); }
</script>
<script>// Copyright 2020 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
'use strict';
// Other constants defined in security_interstitial_page.h.
const SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE =
'displayEnhancedProtectionMessage';
// This sets up the enhanced protection message. function
setupEnhancedProtectionMessage() {
const interstitialType = loadTimeData.getString('type');
if (interstitialType !== 'SAFEBROWSING' && interstitialType !== 'SSL' &&
interstitialType !== 'CAPTIVE_PORTAL') { return;
}
if (!
loadTimeData.getBoolean(SB_DISPLAY_ENHANCED_PROTECTION_MESSAGE)) {
return; }
if ($('enhanced-protection-link')) { if (mobileNav) {
// To make sure the touch area of the link is larger than the
// minimum touch area for accessibility, make the whole block tappable. $('enhanced-
protection-message').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
ON_SETTINGS);
return false; });
} else {
$('enhanced-protection-link').addEventListener('click', function() {
sendCommand(SecurityInterstitialCommandId .CMD_OPEN_ENHANCED_PROTECTI
```

```
ON_SETTINGS);
return false; });
} }
$('enhanced-protection-message').classList.remove('hidden');
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing');
let className = 'ssl-enhanced-protection-message';
if (interstitialType === 'SAFEBROWSING' && !billing) {
className = 'safe-browsing-enhanced-protection-message'; }
$('enhanced-protection-message').classList.add(className); }
</script>
<script>// Copyright 2015 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be
// found in the LICENSE file.
let mobileNav = false;
/**
* For small screen mobile the navigation buttons are moved * below the advanced text.
*/
function onResize() {
const helpOuterBox = document.querySelector('#details');
const mainContent = document.querySelector('#main-content');
const mediaQuery = '(min-width: 240px) and (max-width: 420px) and ' +
'(min-height: 401px), ' +
'(max-height: 560px) and (min-height: 240px) and ' + '(min-width: 421px)';
const detailsHidden = helpOuterBox.classList.contains(HIDDEN_CLASS); const
runnerContainer = document.querySelector('.runner-container');
// Check for change in nav status.
if (mobileNav !== window.matchMedia(mediaQuery).matches) {
mobileNav = !mobileNav;
// Handle showing the top content / details sections according to state. if (mobileNav) {
mainContent.classList.toggle(HIDDEN_CLASS, !detailsHidden);
helpOuterBox.classList.toggle(HIDDEN_CLASS, detailsHidden); if (runnerContainer) {
runnerContainer.classList.toggle(HIDDEN_CLASS, !detailsHidden); }
} else if (!detailsHidden) {
// Non mobile nav with visible details. mainContent.classList.remove(HIDDEN_CLASS);
helpOuterBox.classList.remove(HIDDEN_CLASS); if (runnerContainer) {
runnerContainer.classList.remove(HIDDEN_CLASS); }
} }
}
function setupMobileNav() { window.addEventListener('resize', onResize); onResize();
}
document.addEventListener('DOMContentLoaded', setupMobileNav); </script>
<script>// Copyright 2017 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// This is the shared code for security interstitials. It is used for both SSL // interstitials
```

and Safe Browsing interstitials.

```
/**
 *  @typedef {{
 *  dontProceed: function(),
 *  proceed: function(),
 *  showMoreSection: function(),
 *  openHelpCenter: function(),
 *  openDiagnostic: function(),
 *  reload: function(),
 *  openDateSettings: function(),
 *  openLogin: function(),
 *  doReport: function(),
 *  dontReport: function(),
 *  openReportingPrivacy: function(),
 *  openWhitepaper: function(),
 *  reportPhishingError: function(),
 *  }} */
// eslint-disable-next-line no-var var certificateErrorPageController;
// Should match security_interstitials::SecurityInterstitialCommand /** @enum
{number} */
const SecurityInterstitialCommandId = {
CMD_DONT_PROCEED: 0, CMD_PROCEED: 1,
// Ways for user to get more information CMD_SHOW_MORE_SECTION: 2,
CMD_OPEN_HELP_CENTER: 3, CMD_OPEN_DIAGNOSTIC: 4,
// Primary button actions
CMD_RELOAD: 5, CMD_OPEN_DATE_SETTINGS: 6, CMD_OPEN_LOGIN: 7,
// Safe Browsing Extended Reporting CMD_DO_REPORT: 8,
CMD_DONT_REPORT: 9, CMD_OPEN_REPORTING_PRIVACY: 10,

CMD_OPEN_WHITEPAPER: 11,
// Report a phishing error. CMD_REPORT_PHISHING_ERROR: 12,
// Open enhanced protection settings.
```

```javascript
  CMD_OPEN_ENHANCED_PROTECTION_SETTINGS: 13,
};
const HIDDEN_CLASS = 'hidden';
/**
 * A convenience method for sending commands to the parent page.
 * @param {SecurityInterstitialCommandId} cmd The command to send. */
function sendCommand(cmd) {
if (window.certificateErrorPageController) {
switch (cmd) {
case SecurityInterstitialCommandId.CMD_DONT_PROCEED:
certificateErrorPageController.dontProceed();
break;
case SecurityInterstitialCommandId.CMD_PROCEED:
certificateErrorPageController.proceed();
break;
case SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION:
certificateErrorPageController.showMoreSection();
break;
case SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER:
certificateErrorPageController.openHelpCenter();
break;
case SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC:
certificateErrorPageController.openDiagnostic();
break;
case SecurityInterstitialCommandId.CMD_RELOAD:
certificateErrorPageController.reload();
break;
case SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS:
certificateErrorPageController.openDateSettings();
break;
case SecurityInterstitialCommandId.CMD_OPEN_LOGIN:
certificateErrorPageController.openLogin();
break;
case SecurityInterstitialCommandId.CMD_DO_REPORT:
certificateErrorPageController.doReport();
break;
case SecurityInterstitialCommandId.CMD_DONT_REPORT:
certificateErrorPageController.dontReport(); break;
case SecurityInterstitialCommandId.CMD_OPEN_REPORTING_PRIVACY:
certificateErrorPageController.openReportingPrivacy();
break;
case SecurityInterstitialCommandId.CMD_OPEN_WHITEPAPER:
certificateErrorPageController.openWhitepaper();
break;
case SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR:
certificateErrorPageController.reportPhishingError();
```

```
break;
case
SecurityInterstitialCommandId.CMD_OPEN_ENHANCED_PROTECTION_SETTINGS:
certificateErrorPageController.openEnhancedProtectionSettings();
break; }
return; }
// window.domAutomationController.send(cmd); //
//
}
/**
* Call this to stop clicks on <a href="#"> links from scrolling to the top of * the page (and
possibly showing a # in the link).
*/
function preventDefaultOnPoundLinkClicks() { document.addEventListener('click',
function(e) {
const anchor = findAncestor(/** @type {Node} */ (e.target), function(el) { return
el.tagName === 'A';
});
// Use getAttribute() to prevent URL normalization. if (anchor &&
anchor.getAttribute('href') === '#') {
e.preventDefault(); }
}); }
//
// </script>
<script>// Copyright 2014 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the
LICENSE file.
// This is the shared code for the new (Chrome 37) security interstitials. It is // used for
both SSL interstitials and Safe Browsing interstitials.
let expandedDetails = false; let keyPressState = 0;
/**
* This allows errors to be skippped by typing a secret phrase into the page. * @param
{string} e The key that was just pressed.
*/
function handleKeypress(e) {
// HTTPS errors are serious and should not be ignored. For testing purposes, // other
approaches are both safer and have fewer side-effects.
// See https://goo.gl/ZcZixP for more details.
const BYPASS_SEQUENCE = window.atob('dGhpc2lzdW5zYWZl');
if (BYPASS_SEQUENCE.charCodeAt(keyPressState) === e.keyCode) {
keyPressState++;
if (keyPressState === BYPASS_SEQUENCE.length) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED);
keyPressState = 0; }
} else { keyPressState = 0;
} }
```

```
/**
* This appends a piece of debugging information to the end of the warning.
* * * * * * */
When complete, the caller must also make the debugging div (error-debugging-info)
visible.
@param {string} title The name of this debugging field.
@param {string} value The value of the debugging field.
@param {boolean=} fixedWidth If true, the value field is displayed fixed
width.
function appendDebuggingField(title, value, fixedWidth) {
// The values input here are not trusted. Never use innerHTML on these // values!
const spanTitle = document.createElement('span'); spanTitle.classList.add('debugging-
title');
spanTitle.innerText = title + ': ';
const spanValue = document.createElement('span');
spanValue.classList.add('debugging-content');
if (fixedWidth) {
spanValue.classList.add('debugging-content-fixed-width');
}
spanValue.innerText = value;
const pElem = document.createElement('p'); pElem.classList.add('debugging-content');
pElem.appendChild(spanTitle); pElem.appendChild(spanValue); $('error-debugging-
info').appendChild(pElem);
}
function toggleDebuggingInfo() {
const hiddenDebug = $('error-debugging-info').classList.toggle(HIDDEN_CLASS); $
('error-code').setAttribute('aria-expanded', !hiddenDebug);
}
function setupEvents() {
const overridable = loadTimeData.getBoolean('overridable'); const interstitialType =
loadTimeData.getString('type');
const ssl = interstitialType === 'SSL';
const captivePortal = interstitialType === 'CAPTIVE_PORTAL'; const badClock = ssl &&
loadTimeData.getBoolean('bad_clock'); const lookalike = interstitialType ===
'LOOKALIKE';
const billing =
interstitialType === 'SAFEBROWSING' && loadTimeData.getBoolean('billing'); const
originPolicy = interstitialType === 'ORIGIN_POLICY';
const blockedInterception = interstitialType === 'BLOCKED_INTERCEPTION'; const
legacyTls = interstitialType == 'LEGACY_TLS';
const insecureForm = interstitialType == 'INSECURE_FORM';
const httpsOnly = interstitialType == 'HTTPS_ONLY';
const hidePrimaryButton = loadTimeData.getBoolean('hide_primary_button'); const
showRecurrentErrorParagraph = loadTimeData.getBoolean(
'show_recurrent_error_paragraph');
if (ssl || originPolicy || blockedInterception || legacyTls) { $
```

```
('body').classList.add(badClock ? 'bad-clock' : 'ssl');
if (loadTimeData.valueExists('errorCode')) {
$('error-code').textContent = loadTimeData.getString('errorCode');
$('error-code').classList.remove(HIDDEN_CLASS); }
} else if (captivePortal) { $('body').classList.add('captive-portal');
} else if (billing) { $('body').classList.add('safe-browsing-billing');
} else if (lookalike) { $('body').classList.add('lookalike-url');
} else if (insecureForm) {
$('body').classList.add('insecure-form'); } else if (httpsOnly) {
$('body').classList.add('https-only'); } else {
$('body').classList.add('safe-browsing');
// Override the default theme color. document.querySelector('meta[name=theme-
color]').setAttribute('content',
'rgb(217, 48, 37)'); }
$('icon').classList.add('icon');
if (legacyTls) { $('icon').classList.add('legacy-tls');
}
if (hidePrimaryButton) { $('primary-button').classList.add(HIDDEN_CLASS);
} else {
$('primary-button').addEventListener('click', function() {
switch (interstitialType) { case 'CAPTIVE_PORTAL':
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_LOGIN); break;
case 'SSL':
case 'LEGACY_TLS':
if (badClock) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DATE_SETTINGS); } else
if (overridable) {
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); } else {
sendCommand(SecurityInterstitialCommandId.CMD_RELOAD); }
break;
case 'SAFEBROWSING': case 'ORIGIN_POLICY':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
break;
case 'HTTPS_ONLY': case 'INSECURE_FORM': case 'LOOKALIKE':
sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED); break;
default:
throw 'Invalid interstitial type';
} });
}
if (lookalike || insecureForm || httpsOnly) {
const proceedButton = 'proceed-button'; $
(proceedButton).classList.remove(HIDDEN_CLASS); $(proceedButton).textContent =
loadTimeData.getString('proceedButtonText'); $
(proceedButton).addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
}
```

```
if (lookalike) {
// Lookalike interstitials with a suggested URL have a link in the title:
// "Did you mean <link>example.com</link>?". Handle those clicks. Lookalike //
interstitails without a suggested URL don't have this link.
const dontProceedLink = 'dont-proceed-link';
if ($(dontProceedLink)) {
$(dontProceedLink).addEventListener('click', function(event)
{ sendCommand(SecurityInterstitialCommandId.CMD_DONT_PROCEED);
}); }
}
if (overridable) {
const overrideElement = billing ? 'proceed-button' : 'proceed-link'; // Captive portal page
isn't overridable. $(overrideElement).addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_PROCEED); });
if (ssl) { $(overrideElement).classList.add('small-link');
} else if (billing) { $(overrideElement).classList.remove(HIDDEN_CLASS); $
(overrideElement).textContent =
loadTimeData.getString('proceedButtonText'); }
} else if (!ssl) { $('final-paragraph').classList.add(HIDDEN_CLASS);
}
if (!ssl || !showRecurrentErrorParagraph) { $('recurrent-error-
message').classList.add(HIDDEN_CLASS);
} else { $('body').classList.add('showing-recurrent-error-message');
}
if ($('diagnostic-link')) { $('diagnostic-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_DIAGNOSTIC); });
}
if ($('learn-more-link')) { $('learn-more-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_OPEN_HELP_CENTER); });
}
if (captivePortal || billing || lookalike || insecureForm || httpsOnly) { // Captive portal,
billing, lookalike pages, insecure form, and
// HTTPS only mode interstitials don't have details buttons. $('details-
button').classList.add('hidden');
} else { $('details-button')
.setAttribute(
'aria-expanded', !$('details').classList.contains(HIDDEN_CLASS));
$('details-button').addEventListener('click', function(event) {
const hiddenDetails = $('details').classList.toggle(HIDDEN_CLASS); $('details-
button').setAttribute('aria-expanded', !hiddenDetails);
if (mobileNav) {
// Details appear over the main content on small screens. $('main-
content').classList.toggle(HIDDEN_CLASS, !hiddenDetails);
} else { $('main-content').classList.remove(HIDDEN_CLASS);
}
$('details-button').innerText = hiddenDetails ? loadTimeData.getString('openDetails') :
```

```
loadTimeData.getString('closeDetails');
if (!expandedDetails) {
// Record a histogram entry only the first time that details is opened.
sendCommand(SecurityInterstitialCommandId.CMD_SHOW_MORE_SECTION);
expandedDetails = true;
} });
}
if ($('report-error-link')) { $('report-error-link').addEventListener('click', function(event) {
sendCommand(SecurityInterstitialCommandId.CMD_REPORT_PHISHING_ERROR); })
;
}
if (lookalike) { console.log(
'Chrome has determined that ' + loadTimeData.getString('lookalikeRequestHostname')
+ ' could be fake or fraudulent.\n\n' +
'If you believe this is shown in error please visit ' + 'https://g.co/chrome/lookalike-
warnings');
}
preventDefaultOnPoundLinkClicks(); setupExtendedReportingCheckbox();
setupEnhancedProtectionMessage(); setupSSLDebuggingInfo();
document.addEventListener('keypress', handleKeypress);
}
document.addEventListener('DOMContentLoaded', setupEvents); </script>
</head>
```

————————————————————————

data:image/
png;base64,iVBORw0KGgoAAAANSUhEUgAAAJAAAACQCAMAAADQmBKKAAABTV
BMVEUAAADcRDf/ZmbcRjrjVTn/
gIDdRETdRDzZQzbXQzXWQzbXQjbWQzXZRDbbRDnWQjXWQzXYSDvbRTjcRTjbRT
fcRjfcRTjcRTjdRjncRTfdRTndRTfdRDrbRTjcRDnbRDfbRDjbRjfcRjfbRTjcRTjdRTjbRjjcR
TjcRDjcRjncRTncRTndRDnbRTjcRDfZQzbcRTfgSTncRDfcRjjZQzjcRTfVRDbcRDjcRDj
WQzXeRzvbRDjXRDXXQzbXQzbbRDfeSDvWQjbVQjXIPjLOQDPXQjbCPDDNPzPUQ
TXRQTS5OS7QQDTUQjW3OS7SQTTPQDTFPDHJPjK2OC26OS7HPjHOPzPLPjLMPj
PRQDTGPDHTQTTEPTHLPzPGPTG7Oi/HPTLKPjLTQTXYQza9Oi/
MPzPFPTHDPDHBPDC/OzC+Oy+8Oi/AOzDWQjX////
bRDd3undHAAAAQnRSTlMA2AUWCQIPHj39wvbO8DH64ifqqYFmtrVMc1lKS5x0nY6P
WKqbjYDpZXWCZ1py8Jv9McJXV
+KA9qioPc5l6Y36J7VmcHe8AAAFWUlEQVR4XuzWS4rCQBSG0euz56lSgiaEjHwgGh
AhDnRF3/6HDY1Ia5WPjP4a3LOKY28555xxzjnnnHPOuSyzpPR7vb6lZAUrS8hgB7uBp
aMEKC0Zhz3A/mCpaPjTWCK23GwtCcMjN8ehpWDN3doS8HPi7vRjejX/
1CbX8qA1sdGZB
+eRaW14sjGp8YQnk7EpVQQqE7peCFyupjMnYm4yGVGZ7q1EyTZbEEche2uUbLMIL
5W6t4Zkm22lkm02561c89aQbLNTPpgq3hqSbbbmo1r41rhW8NaAaLMzvjlTvDUg2WzF
lyrBWwOCzc6Jkm12QQcL3Vtlmy3opFC9VbbZJR0tNW
+Vbbahs0b41rhc8FbVZqdEyTb724t5/bYNA3G4e
+80NYl0gGFkvaR779KKZUWuFKe7nIIsT5X//2M5VMZiZB9DQj74xW8ffrwjP90Mb/
07Vf5CbXYJg0BtO4toKS9vhYHGY1vDZg28FQY6tBZls8tYBehwNLTyt1nhrTDQaDQcW

Aux2SJWAxpOBpWMWSvm4q0w0Gg4nFQqFTd/m72HlYBYQJV
+w83bZu9jRaDJYEB4osjJ02aFt8JASUBRq
+PlarMrWBGI8lQajVanXA5kopUcvBUEGrCAWhSoXs3PZtewKhA/
MMbTbcpEa7l4KwwURZSHANnVnGz2CVYGmg6oZ1u1XGy2hNWBCA8BogE1m7Zl
+ShNVMrdW2Wg/v
+Amr2eRYCcGLBZU2+FgcSBESDfdZxdwGbNvBUGihKgnk1OjPAEwS5gsybeCgNNdT
QLyAtqtRCwWQNvhYH4ndjtNnlAnlet1uIQsFl9b4WBpgNyaUCEJ45DwGa1vRUGanU6n
McmB+ZSnlosES3nvm/tUpGm1tFPd5DDAyKFBJGpzRaxSjW5J0o8/MAQ4ZEyKua/b
+0Np175blMERDuaECFBZGqzBaxY9iAjIMbDK01U0OVZxcplE6BIjLzFRixgQDwflCJaX
cC+1ToKyOYHFvCOljPiNmvurTBRI
+oQoGTk2Z1YQyIeiWhlEftWnx8Yf8RcyiMCEkyhic2u4xOWSw9MBBQENTQFI83a
+iL2rdgpJ1rms45mByYzhbDNwt6qTtTlQC7r6FT/CLRQ02ZLWKc8OmK
+LzooCykhKpl4q7p+7B/d0SjNggRbqGOzm1gPqL3PX3niZakOQsenf1PDWzWAxr
+JBtEDQxnnJTISNmvurfBK75t45bORBNGSobcqb9DqBCjdQOl5E370xthbYaDRiIjRDx
KQwJk9a+o2u431gYZERBo/kcBlfvJ/TrSt6K1b+kDUHMkra2V3j5zRlprNbmADILbQ65S/
z2ggyY82zL0VXsdQnnLdhSOKQzWbLWADIMpDgOrd3q958QiigrG3wusYzmNbXmY4s
h+tangrVJ2Dgy97X9v0CmILzzIHcj3ZPTL
+h6DN7mhYR5nxHI4mtKNbLCAmaX9QDDKFO6C36hDttcdJQFGLeTWRIupocGOj62c
Bb9WqesLTFwfm000MQgqz9lDLW
+Hve35HM9Fnqw9HetBkNsF6+Yaet8Jf0+xbka0XbYspSMIg
+5D8/8psnqdYv3qso1vsS9Hy6SaGQ6AYHP9ngLdqllVpiIB8RygRQjGEdOsc4K26RGzk
6YTxjhbDDdzXcfwC8Fbd8glPnR4Y62gBAM/
a1WybfYVNyyUBiZFPXYCAH70GvFW7nFRHH7EgyI8uAd6qXZ7NAqoilG6ZKuBH184D
3qpdAQllWp0p9dE7wFv1q8Y6+njLoPl
+9P4C4K0GRKSjgTyywvoAeKtBxVWU6YhorovcvA14q0HtouwU0Fw/+jzN8w/cQ/
zg6ug2/QAAAABJRU5ErkJggg==
——————————————————————

```
<div class="interstitial-wrapper">
<div id="main-content" class="hidden">
<div class="icon" id="icon"></div> <div id="main-message">
<h1>Your connection is not private</h1>
<p>Attackers might be trying to steal your information from
<strong>accounts.shopify.com</strong> (for example, passwords, messages, or credit
cards). <a href="#" id="learn-more-link">Learn more</a></p> <div id="debugging">
<div id="error-code" class="error-code" role="button" aria-
expanded="false">net::ERR_CERT_DATE_INVALID</div>
<div id="error-debugging-info" class="hidden"><p class="debugging- content"><span
class="debugging-title">Subject: </span><span class="debugging-
content">accounts.shopify.com</span></p><p class="debugging-content"><span
class="debugging-title">Issuer: </span><span class="debugging-content">R3</span></
p><p class="debugging-content"><span class="debugging-title">Expires on: </
span><span class="debugging-content">Nov 13, 2021</span></p><p
class="debugging-content"><span class="debugging-title">Current date: </span><span
class="debugging-content">Oct 5, 2021</span></p><p class="debugging-
content"><span class="debugging-title">PEM encoded chain: </span><span
class="debugging-content debugging-content-fixed-width">-----BEGIN
```

CERTIFICATE-----
MIIF2DCCBMCgAwIBAgISA3lP
+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA
MDIxCzAJBgNVBAYTAlV
TMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQYDVQQD
EwJSMzAeFw
0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNVBAMT
FGFj
Y291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgK
CAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw
gkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw
5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH
w2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/
vtIs
ErLP9noTy1wdKzXJYbu4lT6nVOSs
+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX
MwPm/Opq3qk3Bnl5uZgyV
+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/
BAQD
AgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBg
NVHRMBAf8EAjAA
MB0GA1UdDgQWBBR/TRz5xF8Ma/
wcElldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX
t1hWy65QCUDmH6+dixTCxjBVB
ggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGFWh0
dHA6Ly9yMy5vLmxlbmN
yLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkubGVu
Y3Iub3JnLzCByQYDV
R0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91
ZC5jb22CFGFjY291bnRzLnNob3B
pZnkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w
aWZ5Y2xvdWQuY29tgiZpZGVudGl0
eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk
LmNvbYIiaWRlbnRpdHktbWVyY2h
hbnQuc2hvcGlmeWNsb3VkLmNvbYIZaWRlbnRp
dHkuc2hvcGlmeWNsb3VkLmNv
bTBMBgNVHSAERTBDMAgGBmeBDAECATA3BgsrBgEE
AYLfEwEBATAoMCYGC
CsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0Lm9y
ZzCCAQMGCisGA
QQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj+KMDa5oK+
2MsxtT/
TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf
+YE3BwY4lHwrPfCyx
HBfmiWZVYY3syP6PMtQClCbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14lTkrE
AHYAfT7y+l//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA
RzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfgIhAOGXW8pNxh
+q
AgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQ
BcKanD
DWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWIfOU
LxGoEd/k
SA0t9X94AFhu+FhBQnrssUH/D
+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
dWfgQcEhx3HInYVT
+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h
Z8Y/kug
+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5
MVXdjQTeKuzilyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3l7Z70dX1sEwUXQR8Wr
/+lfVfwGqqjAuskK
-----END
CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/
bWLiWnX1owDQYJKoZIhvcNAQELBQAw
TzELMAkGA1UEBhMCVVMxKTAnBgN
VBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQD
EwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDAwMDAw
WhcNMjUwOTE1MTYw
MDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChMNTGV0J3Mg
RW5jcnl
wdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEK

AoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r+XXIo9cP
R5QUVTVXjJ6oojkZ9YI8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCUJCWH+jdx
sxPnHKzhm+/b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
NHz6a4uPVymZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
Z3Vms/EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG
/kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
AYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/wQIMAYB
Af8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrlAJQOYfr52LFMLGMB8GA1UdIwQYMBaA
FHm0WeZ7tuXkAXOACIjlGlj26ZtuMDIGCCsGAQUFBwEBBCYwJDAiBggrBgEFBQcw
AoYWaHR0cDovL3gxLmkubGVuY3lub3JnLzAnBgNVHR8EIDAeMBygGqAYhhZodHRw
Oi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCAYGZ4EMAQIBMA0GCysGAQQB
gt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQCFyk5HPqP3hUSFvNVneLKYY611TR6W
PTNlclQtgaDqw+34IL9fzLdwALduO/ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
ikfmZW4/RvTZ8M6UK+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
CkcheAmCJ8MqyJu5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh+BIm
lJNXoB1lBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/yFOGwXD9RiX8F6sw6W4
avAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f+lQKc0t8DQYzk1OXVu8rp2
yJMC6alLbBfODALZvYH7n7do1AZls4I9d1P4jnkDrQoxB3UqQ9hVl3LEKQ73xF1O
yK5GhDDX8oVfGKF5u+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90Ids
hCExroL1+7mryIkXPeFM5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+
HlUjr8gRsl3qfJOQFy/9rKIJR0Y/8Omwt/8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
MldlTTKB3zhThV1+XWYp6rjd5JW1zbVWEkLNxE7GJThEUG3szgBVGP7pSWTUTsqX
nLRbwHOoq7hHwg==
-----ENDCERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIFYDCCBEigAwIBAgIQQAF3ITfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFADA/
MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBAMT
DkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMDE4MTQwM1ow
TzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
cmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwggIiMA0GCSqGSIb3DQEB
AQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHIe+3LffOJCMbjzmV6B493XC
ov71am72AE8o295ohmxEk7axY/0UEmu/H9LqMZshftEzPLpI9d1537O4/xLxIZpL
wYqGcWlKZmZsj348cL+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlIl1PoOK5PCm+D
LtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P+1dy70sntK
4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5
bHbvO5BieebbpJovJsXQEOEO3tkQjhb7t/eo98flAgeYjzYllefiN5YNNnWe+w5y
sR2bvAP5SQXYgd0FtCrWQemsAXaVCg/Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ
Xmidf3LWicUGQSk+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4
FQsDj43QLwWyPntKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
SLeCO5imfWCKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/H2ql
PRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/

7xk9TND
TwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
SwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUF
BzAChi9odHRwOi8vYXBwcy5pZGVudHJ1
c3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgz
LnA3YzAfBgNVHSMEGDAWgBTEp7Gkeyxx
+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
ATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3Qte
DEubGV0c2VuY3J5cHQu
b3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9
jcmwuaWRlbnRydXN0LmNvbS9E
U1RST09UQ0FYM0NSTC5jcmwwHQYDVR0OB
BYEFHm0WeZ7tuXkAXOACIjlGlj26Ztu
MA0GCSqGSIb3DQEBCwUAA4IBAQAKcw
Bslm7/DlLQrt2M51oGrS+o44+/
yQoDFVDC
5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WIHAQU7
G75K/
QosMrAdSW
9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
WCLKTVXkcGdtwIfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6U
SbW1FzXSLr9O
he8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE
+FiYruq9RUWhiF1myv4Q6W
+CyBFC
Dfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5

END CERTIFICATE-----
-----BEGIN CERTIFICATE-----

MIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/

MSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNV
BAMT
DkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE
0MDExNVow
PzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ
28uMRcwFQYDVQQD
Ew5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNA
QEBBQADggEPADCCAQoCggEB
AN
+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O
rz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
OLl5CjH9
UL2AZd
+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
xiqKqy69cK3FCx
olkHRyxXtqqzTWMIn/5WgTe1QLyNau7Fqckh49ZLOMxt+/
yUFw
7BZy1SbsOFU5Q9D8/
RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
aeQQmxkqtilX4+U9m
5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV
HQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG
SIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNI+DT69
ikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX
+5v3gTt23ADq1cEmv8uXr
AvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbu
mR9YbK
+rlmM6pZW87ipxZz
R8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZImlJnt1ir/
md2cXjbDaJWFBM5
JDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo
Ob8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ<b
r>-----END CERTIFICATE-----
</p><p class="debugging-content">Certificate Transparency: <span class="debugging-
content">

SCT Cloudflare 'Nimbus2021' Log (Embedded in certificate,

Verified)

SCT Google 'Xenon2021' log (Embedded in certificate, Verified)</p></div>
</div>
</div>
<div id="recurrent-error-message" class="hidden">
Warnings may be common while websites update their security. This should improve soon.
</div>
<div id="extended-reporting-opt-in" class="hidden">
<label>
<div class="checkboxes">
<input type="checkbox" id="opt-in-checkbox">

</div>
Help improve security on the web for everyone by sending URLs of some pages you visit, limited system information, and some page content to Google. Privacy policy
</label>
</div>
<div id="enhanced-protection-message" class="hidden">
<label>
<div class="icon"></div>
<div>To get Chrome's highest level of security, turn on enhanced protection</div> </label>
</div>
</div>
<div class="nav-wrapper">
<button id="primary-button">Reload</button>
<button id="proceed-button" class="secondary-button small-link hidden"></button>
<button id="details-button" class="secondary-button small-link" aria-expanded="true">Hide advanced</button> </div>
<div id="details" class="">
<p>accounts.shopify.com normally uses encryption to protect your information. When Chrome tried to connect to accounts.shopify.com this time, the website sent back unusual and incorrect credentials. This may happen when an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.</p>
<p id="final-paragraph">You cannot visit accounts.shopify.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.</p>
</div> </div>
————————————————————————————
<style>/* Copyright 2014 The Chromium Authors. All rights reserved.
  * Use of this source code is governed by a BSD-style license that can be

-     * found in the LICENSE file. */

/* This file is dynamically processed by a C++ data source handler to fill in
-     * some per-platform/locale styles that dramatically alter the page. This is

-     * done to reduce flicker, as JS may not run before the page is rendered.
  *

-     * There are two ways to include this stylesheet:

-     * 1. via its chrome://resources/ URL in HTML, i.e.: *

-     * <link rel="stylesheet" href="chrome://resources/css/text_defaults.css">
  *

-     * 2. via the webui::AppendWebUICSSTextDefaults() method to directly append it

-     * to an HTML string.

-     * Otherwise its placeholders won't be expanded. */
  body {
  font-family: system-ui, sans-serif; font-size: 75%;
  }
  button {
  font-family: system-ui, sans-serif;
  } </style>

———————————————————————————

/**
-     * @fileoverview This file defines a singleton which provides access to all data

-     * that is available as soon as the page's resources are loaded (before DOM

-     * content has finished loading). This data includes both localized strings and

-     * any data that is important to have ready from a very early stage (e.g. things

-     * that must be displayed right away). *

-     * Note that loadTimeData is not guaranteed to be consistent between page

-     * refreshes (https://crbug.com/740629) and should not contain values that might

-     * change if the page is re-opened later. */
  /** @type {!LoadTimeData} */
  // eslint-disable-next-line no-var var loadTimeData;
  class LoadTimeData { constructor() {
  /** @type {?Object} */
  this.data_ = null; }
  /**
  -     * Sets the backing object. *

  -     * Note that there is no getter for |data_| to discourage abuse of the form: *

  -     * var value = loadTimeData.data()['key']; *

  -     * @param {Object} value The de-serialized page data. */
    set data(value) {
    expect(!this.data_, 'Re-setting data.'); this.data_ = value;
    }
    /**
    * @param {string} id An ID of a value that might exist.

* @return {boolean} True if |id| is a key in the dictionary.
*/ valueExists(id) {
return id in this.data_; }
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value = this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**
* As above, but also makes sure that the value is a string.
* @param {string} id The key that identifies the desired string. * @return {string} The corresponding string value.

```
*/
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
```

- * Returns a formatted localized string where $1 to $9 are replaced by the

- * second to the tenth argument.

- * @param {string} id The ID of the string we want.

- * @param {...(string|number)} var_args The extra values to include in
  the

- * formatted output.

- * @return {string} The formatted string. */

```
getStringF(id, var_args) {
const value = this.getString(id); if (!value) {
return ''; }
const args = Array.prototype.slice.call(arguments); args[0] = value;
return this.substituteString.apply(this, args);

}
/**
```

- * Returns a formatted localized string where $1 to $9 are replaced by the

- * second to the tenth argument. Any standalone $ signs must be
  escaped as

- * $$.

```
* * * * * */
@param {string} label The label to substitute through. This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted
output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
} /**
* * * * * * * * * */
Returns a formatted string where $1 to $9 are replaced by the second to tenth
```

argument, split apart into a list of pieces describing how the substitution was performed. Any standalone $ signs must be escaped as $$. @param {string} label A localized string to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted string pieces.
getSubstitutedStringPieces(label, var_args) {
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9 pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) || []).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
* As above, but also makes sure that the value is a boolean.
* @param {string} id The key that identifies the desired boolean. * @return {boolean} The corresponding boolean value.
*/
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type {boolean} */ (value);
}
/**
* As above, but also makes sure that the value is an integer.
* @param {string} id The key that identifies the desired number. * @return {number} The corresponding number value.
*/
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type {number} */ (value);
}
/**
* Override values in loadTimeData with the values found in |replacements|. * @param {Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) { expect(

```
typeof replacements === 'object',
'Replacements must be a dictionary object.'); for (const key in replacements) {
this.data_[key] = replacements[key]; }
}
/**
* Reset loadTimeData's data. Should only be used in tests.
* @param {?Object} newData The data to restore to, when null restores to
* unset state.
*/
resetForTesting(newData = null) {
this.data_ = newData; }
/**
* @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
* Checks condition, throws error message if expectation fails.
* @param {*} condition The condition to check for truthiness.
* @param {string} message The message to display if the check fails. */
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
} }
/**
* Checks that the given value has the given type.
* @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
* @param {string} type The type we expect |value| to be.
*/
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');</script>
——————————————————————————
<script>loadTimeData.data = {"bad_clock":false,"closeDetails":"Hide advanced","ct":"\n
\nSCT Cloudflare 'Nimbus2021' Log (Embedded in certificate, Verified)\n\nSCT Google
'Xenon2021' log (Embedded in certificate, Verified)","currentDate":"Oct 5,
2021","displayEnhancedProtectionMessage":false,"displaycheckbox":false,"enhancedPr
otectionMessage":"To get Chrome's highest level of security, \u003Ca href=\"#\" id=
\"enhanced-protection-link\">turn on enhanced protection\u003C/
```

a>","errorCode":"net::ERR_CERT_DATE_INVALID","errorType":1,"expirationDate":"Nov 13, 2021","explanationParagraph":"accounts.shopify.com normally uses encryption to protect your information. When Chrome tried to connect to accounts.shopify.com this time, the website sent back unusual and incorrect credentials. This may happen when an attacker is trying to pretend to be accounts.shopify.com, or a Wi-Fi sign-in screen has interrupted the connection. Your information is still secure because Chrome stopped the connection before any data was exchanged.","finalParagraph":"You cannot visit accounts.shopify.com right now because the website uses HSTS. Network errors and attacks are usually temporary, so this page will probably work later.","fontfamily":"system-ui, sans-serif","fontsize":"75%","heading":"Your connection is not private","hide_primary_button":false,"issuer":"R3","language":"en","openDetails":"Advanced","optInLink":"Help improve security on the web for everyone by sending \u003Ca href=\"#\" id=\"whitepaper-link\">URLs of some pages you visit, limited system information, and some page content\u003C/a> to Google. \u003Ca id=\"privacy-link\" href=\"#\">Privacy policy\u003C/a>","overridable":false,"pem":"-----BEGIN CERTIFICATE-----\nMIIF2DCCBMCgAwIBAgISA3lP

+wuRtzmomjRFbuWpyrsCMA0GCSqGSIb3DQEBCwUA
\nMDIxCzAJBgNVBAYTAlVTMRYwFAYDVQQKEw1MZXQncyBFbmNyeXB0MQswCQY DVQQD
\nEwJSMzAeFw0yMTA4MTUxNjE3MDNaFw0yMTExMTMxNjE3MDFaMB8xHTAbBgNV BAMT
\nFGFjY291bnRzLnNob3BpZnkuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
\nCgKCAQEAmNO6kENQqqUM0WgozXecZGcPzrIyEbrU/
N7mdStXBNjXN9WAEbLSTGrw\ngkyShj1dbDXVJudJ7YdQXJrl/
KJbm87u8a1HEZl4Dmj1FxBIAd+Gs1RsFdQd0nEw\n5Zgi0Sojj6d87SEqpS2i0Vz6V
+hTrbuy+XXP5+J8piNiRp2lkrZQkWKaifZaLyvH\nw2S2OFtEJxH3SSoKfsA/
ST4ZPRDFIppsXgmpFf7sU5u1E3XckNJxl220Q29/vtIs
\nErLP9noTy1wdKzXJYbu4lT6nVOSs+TGLQBcBCfNQyaG19jMrL299G4QyNy2h6uWX
\nMwPm/Opq3qk3BnI5uZgyV+mx7EMe1QIDAQABo4IC+TCCAvUwDgYDVR0PAQH/ BAQD
\nAgWgMB0GA1UdJQQWMBQGCCsGAQUFBwMBBggrBgEFBQcDAjAMBgNVHRMB Af8EAjAA\nMB0GA1UdDgQWBBR/TRz5xF8Ma/
wcEIldjg8fyl3L3zAfBgNVHSMEGDAWgBQULrMX
\nt1hWy65QCUDmH6+dixTCxjBVBggrBgEFBQcBAQRJMEcwIQYIKwYBBQUHMAGGGF Wh0\ndHA6Ly9yMy5vLmxlbmNyLm9yZzAiBggrBgEFBQcwAoYWaHR0cDovL3IzLmkub GVu\nY3Iub3JnLzCByQYDVR0RBIHBMIG
+gh5hY2NvdW50cy1kZW1vLnNob3BpZnljbG91\nZC5jb22CFGFjY291bnRzLnNob3BpZ nkuY29tgh9jdXN0b21lcnMtZGVtby5zaG9w
\naWZ5Y2xvdWQuY29tgiZpZGVudGl0eS1tYWluLWNsdXN0ZXIuc2hvcGlmeWNsb3Vk
\nLmNvbYIiaWRlbnRpdHktbWVyY2hhbnQuc2hvcGlmeWNsb3VkLmNvYIZaWRlbnRp
\ndHkuc2hvcGlmeWNsb3VkLmNvbTBMBgNVHSAERTBDMAgGBmeBDAECATA3Bgsr BgEE
\nAYLfEwEBATAoMCYGCCsGAQUFBwIBFhpodHRwOi8vY3BzLmxldHNlbmNyeXB0L m9y\nZzCCAQMGCisGAQQB1nkCBAIEgfQEgfEA7wB1AESUZS6w7s6vxEAH2Kj
+KMDa5oK+\n2MsxtT/TM5a1toGoAAABe0rRIFkAAAQDAEYwRAIgcu9G4Kf

+YE3BwY4lHwrPfCyx\nHBfmiWZVYY3syP6PMtQCICbrsrvVPy0eYoa
+way7bJOxuRAbwT4Sfd4Nd14ITkrE\nAHYAfT7y+I//
iFVoJMLAyp5SiXkrxQ54CX8uapdomX4i8NcAAAF7StEgegAABAMA\nRzBFAiBQFP/
Jq9s0RX2IddzPRWorKmK2vHb0gBjdeR4DafmJfglhAOGXW8pNxh+q
\nAgkEhILco7khwJUOAcDqcKCl9HS9FOJ8MA0GCSqGSIb3DQEBCwUAA4IBAQBcKa
nD\nDWicj0QRpqqbMyHi1uwU0vJoOuCXAB5TMevD1OGhl2HGfV8blgxWlfOULxGoEd/
k\nSA0t9X94AFhu+FhBQnrssUH/D+xQFKSbaZcQlB5j1nEFNdtpokxW5JyhjS5MXiYO
\ndWfgQcEhx3HInYVT+Z1CpNslgCSOaoOawx+b1y1/Kj5ZlEId2rZBlOxh+/ifPP9h\nZ8Y/
kug+O2Vu7xfPWn4RtVeG39FpK3FcbOxkTA09K1CHtPQf+KZ/
Ou5BcLn8BJi5\nMVXdjQTeKuziIyV7QM+dAxHn3/
reMXmlGengm7njl9K2P3I7Z70dX1sEwUXQR8Wr\n/+lfVfwGqqjAuskK\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFFjCCAv6gAwIBAgIRAJErCErPDBinU/bWLiWnX1owDQYJKoZIhvcNAQELBQAw
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwHhcNMjAwOTA0MDA
wMDAw
\nWhcNMjUwOTE1MTYwMDAwWjAyMQswCQYDVQQGEwJVUzEWMBQGA1UEChM
NTGV0J3Mg
\nRW5jcnlwdDELMAkGA1UEAxMCUjMwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAw
ggEK\nAoIBAQC7AhUozPaglNMPEuyNVZLD+ILxmaZ6QoinXSaqtSu5xUyxr45r
+XXIo9cP
\nR5QUVTVXjJ6oojkZ9Yl8QqlObvU7wy7bjcCwXPNZOOftz2nwWgsbvsCUJCWH+jdx
\nsxPnHKzhm+/b5DtFUkWWqcFTzjTIUu61ru2P3mBw4qVUq7ZtDpelQDRrK9O8Zutm
\nNHz6a4uPVymZ+DAXXbpyb/uBxa3Shlg9F8fnCbvxK/eG3MHacV3URuPMrSXBiLxg
\nZ3Vms/EY96Jc5lP/Ooi2R6X/ExjqmAl3P51T+c8B5fWmcBcUr2Ok/5mzk53cU6cG\n/
kiFHaFpriV1uxPMUgP17VGhi9sVAgMBAAGjggEIMIIBBDAOBgNVHQ8BAf8EBAMC
\nAYYwHQYDVR0lBBYwFAYIKwYBBQUHAwIGCCsGAQUFBwMBMBIGA1UdEwEB/
wQIMAYB
\nAf8CAQAwHQYDVR0OBBYEFBQusxe3WFbLrIAJQOYfr52LFMLGMB8GA1UdIwQY
MBaA
\nFHm0WeZ7tuXkAXOACIjIGlj26ZtuMDIGCCsGAQUFBwEBBCYwJDAiBggrBgEFBQc
w
\nAoYWaHR0cDovL3gxLmkubGVuY3Iub3JnLzAnBgNVHR8EIDAeMBygGqAYhhZodH
Rw
\nOi8veDEuYy5sZW5jci5vcmcvMCIGA1UdIAQbMBkwCAYGZ4EMAQIBMA0GCysGAQ
QB
\ngt8TAQEBMA0GCSqGSIb3DQEBCwUAA4ICAQCFyk5HPqP3hUSFvNVneLKYY611T
R6W\nPTNlclQtgaDqw+34IL9fzLdwALduO/ZelN7kIJ+m74uyA+eitRY8kc607TkC53wl
\nikfmZW4/RvTZ8M6UK+5UzhK8jCdLuMGYL6KvzXGRSgi3yLgjewQtCPkIVz6D2QQz
\nCkcheAmCJ8MqyJu5zlzyZMjAvnnAT45tRAxekrsu94sQ4egdRCnbWSDtY7kh+BIm
\nnlJNXoB1IBMEKIq4QDUOXoRgffuDghje1WrG9ML+Hbisq/
yFOGwXD9RiX8F6sw6W4\navAuvDszue5L3sz85K+EC4Y/wFVDNvZo4TYXao6Z0f
+lQKc0t8DQYzk1OXVu8rp2\nyJMC6alLbBfODALZvYH7n7do1AZls4l9d1P4jnkDrQoxB3
UqQ9hVl3LEKQ73xF1O\nyK5GhDDX8oVfGKF5u
+decIsH4YaTw7mP3GFxJSqv3+0lUFJoi5Lc5da149p90lds
\nhCExroL1+7mryIkXPeFM5TgO9r0rvZaBFOvV2z0gp35Z0+L4WPlbuEjN/lxPFin+

\nHlUjr8gRsl3qfJOQFy/9rKlJR0Y/8Omwt/8oTWgy1mdeHmmjk7j1nYsvC9JSQ6Zv
\nMldlTTKB3zhThV1+XWYp6rjd5JW1zbVWEkLNxE7GJThEUG3szgBVGP7pSWTUTsq
X\nnLRbwHOoq7hHwg==\n-----END CERTIFICATE-----\n-----BEGIN CERTIFICATE-----
\nMIIFYDCCBEigAwIBAgIQQAF3lTfU6UK47naqPGQKtzANBgkqhkiG9w0BAQsFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT
\nDkRTVCBSb290IENBIFgzMB4XDTIxMDEyMDE5MTQwM1oXDTI0MDkzMDE4MTQw
M1ow
\nTzELMAkGA1UEBhMCVVMxKTAnBgNVBAoTIEludGVybmV0IFNlY3VyaXR5IFJlc2Vh
\ncmNoIEdyb3VwMRUwEwYDVQQDEwxJU1JHIFJvb3QgWDEwggIiMA0GCSqGSIb3D
QEB\nAQUAA4ICDwAwggIKAoICAQCt6CRz9BQ385ueK1coHIe
+3LffOJCMbjzmV6B493XC\nov71am72AE8o295ohmxEk7axY/0UEmu/
H9LqMZshftEzPLpI9d1537O4/xLxIZpL\nwYqGcWlKZmZsj348cL
+tKSIG8+TA5oCu4kuPt5l+lAOf00eXfJlII1PoOK5PCm+D
\nLtFJV4yAdLbaL9A4jXsDcCEbdfIwPPqPrt3aY6vrFk/CjhFLfs8L6P+1dy70sntK
\n4EwSJQxwjQMpoOFTJOwT2e4ZvxCzSow/
iaNhUd6shweU9GNx7C7ib1uYgeGJXDR5\nbHbvO5BieebbpJovJsXQEOEO3tkQjhb7t/
eo98flAgeYjzYllefiN5YNNnWe+w5y\nsR2bvAP5SQXYgd0FtCrWQemsAXaVCg/
Y39W9Eh81LygXbNKYwagJZHduRze6zqxZ\nXmidf3LWicUGQSk
+WT7dJvUkyRGnWqNMQB9GoZm1pzpRboY7nn1ypxIFeFntPlF4\nFQsDj43QLwWyPn
tKHEtzBRL8xurgUBN8Q5N0s8p0544fAQjQMNRbcTa0B7rBMDBc
\nSLeCO5imfWCKoqMpgsy6vYMEG6KDA0Gh1gXxG8K28Kh8hjtGqEgqiNx2mna/H2ql
\nPRmP6zjzZN7IKw0KKP/32+IVQtQi0Cdd4Xn+GOdwiK1O5tmLOsbdJ1Fu/7xk9TND
\nTwIDAQABo4IBRjCCAUIwDwYDVR0TAQH/BAUwAwEB/
zAOBgNVHQ8BAf8EBAMCAQYw
\nSwYIKwYBBQUHAQEEPzA9MDsGCCsGAQUFBzAChi9odHRwOi8vYXBwcy5pZGVu
dHJ1\nc3QuY29tL3Jvb3RzL2RzdHJvb3RjYXgzLnA3YzAfBgNVHSMEGDAWgBTEp7G
keyxx\n+tvhS5B1/8QVYIWJEDBUBgNVHSAETTBLMAgGBmeBDAECATA/
BgsrBgEEAYLfEwEB
\nATAwMC4GCCsGAQUFBwIBFiJodHRwOi8vY3BzLnJvb3QteDEubGV0c2VuY3J5cHQu
\nb3JnMDwGA1UdHwQ1MDMwMaAvoC2GK2h0dHA6Ly9jcmwuaWRlbnRydXN0LmNv
bS9E
\nU1RST09UQ0FYM0NTC5jcmwwHQYDVR0OBBYEFHm0WeZ7tuXkAXOACIjIGlj26
Ztu\nMA0GCSqGSIb3DQEBCwUAA4IBAQAKcwBslm7/DILQrt2M51oGrS+o44+/
yQoDFVDC
\n5WxCu2+b9LRPwkSICHXM6webFGJueN7sJ7o5XPWioW5WlHAQU7G75K/
QosMrAdSW\n9MUgNTP52GE24HGNtLi1qoJFlcDyqSMo59ahy2cI2qBDLKobkx/
J3vWraV0T9VuG
\nWCLKTVXkcGdtwlfFRjlBz4pYg1htmf5X6DYO8A4jqv2Il9DjXA6USbW1FzXSLr9O
\nhe8Y4IWS6wY7bCkjCWDcRQJMEhg76fsO3txE+FiYruq9RUWhiF1myv4Q6W
+CyBFC\nDfvp7OOGAN6dEOM4+qR9sdjoSYKEBpsr6GtPAQw4dy753ec5\n-----END
CERTIFICATE-----\n-----BEGIN CERTIFICATE-----\nMIIDSjCCAjKgAwIBAgIQRK
+wgNajJ7qJMDmGLvhAazANBgkqhkiG9w0BAQUFADA/
\nMSQwIgYDVQQKExtEaWdpdGFsIFNpZ25hdHVyZSBUcnVzdCBDby4xFzAVBgNVBA
MT

\nDkRTVCBSb290IENBIFgzMB4XDTAwMDkzMDIxMTIxOVoXDTIxMDkzMDE0MDExN
Vow
\nPzEkMCIGA1UEChMbRGlnaXRhbCBTaWduYXR1cmUgVHJ1c3QgQ28uMRcwFQYD
VQQD
\nEw5EU1QgUm9vdCBDQSBYMzCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQ
oCggEB\nAN+v6ZdQCINXtMxiZfaQguzH0yxrMMpb7NnDfcdAwRgUi+DoM3ZJKuM/
IUmTrE4O\nrz5Iy2Xu/
NMhD2XSKtkyj4zl93ewEnu1lcCJo6m67XMuegwGMoOifooUMM0RoOEq
\nOLl5CjH9UL2AZd+3UWODyOKIYepLYYHsUmu5ouJLGiifSKOeDNoJjj4XLh7dIN9b
\nxxiqKqy69cK3FCxolkHRyxXtqqzTWMln/5WgTe1QLyNau7Fqckh49ZLOMxt+/yUFw
\n7BZy1SbsOFU5Q9D8/RhcQPGX69Wam40dutolucbY38EVAjqr2m7xPi71XAicPNaD
\naeQQmxkqtilX4+U9m5/wAl0CAwEAAaNCMEAwDwYDVR0TAQH/BAUwAwEB/
zAOBgNV\nHQ8BAf8EBAMCAQYwHQYDVR0OBBYEFMSnsaR7LHH62+FLkHX/
xBVghYkQMA0GCSqG\nSIb3DQEBBQUAA4IBAQCjGiybFwBcqR7uKGY3Or
+Dxz9LwwmglSBd49lZRNI+DT69\nikugdB/
OEIKcdBodfpga3csTS7MgROSR6cz8faXbauX+5v3gTt23ADq1cEmv8uXr
\nAvHRAosZy5Q6XkjEGB5YGV8eAlrwDPGxrancWYaLbumR9YbK+rlmM6pZW87ipxZz
\nR8srzJmwN0jP41ZL9c8PDHIyh8bwRLtTcm1D9SZlmlJnt1ir/
md2cXjbDaJWFBM5\nJDGFoqgCWjBH4d1QB7wCCZAA62RjYJsWvIjJEubSfZGL
+T0yjWW06XyxV3bqxbYo\nOb8VZRzI9neWagqNdwvYkQsEjgfbKbYK7p2CNTUQ\n-----
END CERTIFICATE-----\n","primaryButtonText":"Reload","primaryParagraph":"Attackers might be trying to steal your information from \u003Cstrong>accounts.shopify.com \u003C/strong> (for example, passwords, messages, or credit cards). \u003Ca href=\"#\" id=\"learn-more-link\">Learn more\u003C/a>","recurrentErrorParagraph":"Warnings may be common while websites update their security. This should improve soon.","show_recurrent_error_paragraph":false,"subject":"accounts.shopify.com","tabTitle":"Privacy error","textdirection":"ltr","type":"SSL"};</script>
————————————————————————
/**
* @fileoverview Assertion support. */
/**
- * Verify |condition| is truthy and return |condition| if so.
- * @template T
- * @param {T} condition A condition to check for truthiness. Note that this
- * may be used to test whether a value is defined or not, and we don't want
- * to force a cast to Boolean.
- * @param {string=} opt_message A message to show on failure.

- * @return {T} A non-null !condition!.

- * @closurePrimitive {asserts.truthy}

- * @suppress {reportUnknownTypes} because T is not sufficiently
    constrained. */
    /* #export */ function assert(condition, opt_message) { if (!condition) {
    let message = 'Assertion failed'; if (opt_message) {
    message = message + ': ' + opt_message; }
    const error = new Error(message); const global = function() {
    const thisOrSelf = this || self;
    /** @type {boolean} */ thisOrSelf.traceAssertionsForTesting; return thisOrSelf;
    }();
    if (global.traceAssertionsForTesting) {
    console.warn(error.stack); }
    throw error; }
    return condition; }
    /**

* * * * * * * * *
Call this from places in the code that should never be reached.
For example, handling all the values of enum with a switch() like this:
function getValueFromEnum(enum) { switch (enum) {
case ENUM_FIRST_OF_TWO: return first
case ENUM_LAST_OF_TWO:
* return last; *}

- * assertNotReached();

- * return document; *}
    *

* This code should only be hit in the case of serious programmer error or
* * * * */
unexpected input.
@param {string=} opt_message A message to show when this is hit. @closurePrimitive
{asserts.fail}
/* #export */ function assertNotReached(opt_message) { assert(false, opt_message ||
'Unreachable code hit');
}
/**
* @param {*} value The value to check.
* @param {function(new: T, ...)} type A user-defined constructor.
* @param {string=} opt_message A message to show when this is hit. * @return {T}
* @template T
*/

```
/* #export */ function assertInstanceof(value, type, opt_message) {
// We don't use assert immediately here so that we avoid constructing an error //
message if we don't have to.
if (!(value instanceof type)) {
assertNotReached(
opt_message ||
'Value ' + value + ' is not a[n] ' + (type.name || typeof type));
}
return value; }
/* #ignore */ console.warn('crbug/1173575, non-JS module files deprecated.');
```

——————————————————————————

VM2261:2134 crbug/1173575, non-JS module files deprecated. (anonymous) @ VM2261:2134
debugger:///VM2261

```
// Copyright (c) 2012 The Chromium Authors. All rights reserved.
// Use of this source code is governed by a BSD-style license that can be // found in the LICENSE file.
/**
```

- * @fileoverview This file defines a singleton which provides access to all data

- * that is available as soon as the page's resources are loaded (before DOM

- * content has finished loading). This data includes both localized strings and

- * any data that is important to have ready from a very early stage (e.g. things

- * that must be displayed right away). *

- * Note that loadTimeData is not guaranteed to be consistent between page

- * refreshes (https://crbug.com/740629) and should not contain values that might

- * change if the page is re-opened later. */

```
/** @type {!LoadTimeData} */
// eslint-disable-next-line no-var var loadTimeData;
class LoadTimeData { constructor() {
/** @type {?Object} */
this.data_ = null; }
/**
```

  - * Sets the backing object. *

- * Note that there is no getter for |data_| to discourage abuse of the form: *

- * var value = loadTimeData.data()['key']; *

- * @param {Object} value The de-serialized page data. */
```
set data(value) {
expect(!this.data_, 'Re-setting data.'); this.data_ = value;
}
/**
* @param {string} id An ID of a value that might exist.
* @return {boolean} True if |id| is a key in the dictionary. */
valueExists(id) {
return id in this.data_;

}
/**
* Fetches a value, expecting that it exists.
* @param {string} id The key that identifies the desired value. * @return {*} The
corresponding value.
*/
getValue(id) {
expect(this.data_, 'No data. Did you remember to include strings.js?'); const value =
this.data_[id];
expect(typeof value !== 'undefined', 'Could not find value for ' + id); return value;
}
/**
* As above, but also makes sure that the value is a string.
* @param {string} id The key that identifies the desired string. * @return {string} The
corresponding string value.
*/
getString(id) {
const value = this.getValue(id); expectIsType(id, value, 'string'); return /** @type {string}
*/ (value);
}
/**
```

- * Returns a formatted localized string where $1 to $9 are replaced by the

- * second to the tenth argument.

- * @param {string} id The ID of the string we want.

- * @param {...(string|number)} var_args The extra values to include in the

-         * formatted output.

-         * @return {string} The formatted string. */

```
getStringF(id, var_args) {
const value = this.getString(id); if (!value) {
return ''; }
const args = Array.prototype.slice.call(arguments); args[0] = value;
return this.substituteString.apply(this, args);
}
/**
* Returns a formatted localized string where $1 to $9 are replaced by the
```

```
* * * * * * * */
second to the tenth argument. Any standalone $ signs must be escaped as $$.
@param {string} label The label to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the
formatted output.
@return {string} The formatted string.
substituteString(label, var_args) {
const varArgs = arguments;
return label.replace(/\$(.|$|\n)/g, function(m) {
expect(m.match(/\$[$1-9]/), 'Unescaped $ found in localized string.');
return m === '$$' ? '$' : varArgs[m[1]]; });
} /**
* * * * * * * * * */
Returns a formatted string where $1 to $9 are replaced by the second to tenth
argument, split apart into a list of pieces describing how the substitution was performed.
Any standalone $ signs must be escaped as $$. @param {string} label A localized string
to substitute through.
This is not an resource ID.
@param {...(string|number)} var_args The extra values to include in the
formatted output.
@return {!Array<!{value: string, arg: (null|string)}>} The formatted
string pieces.
getSubstitutedStringPieces(label, var_args) {
const varArgs = arguments;
// Split the string by separately matching all occurrences of $1-9 and of // non $1-9
pieces.
const pieces = (label.match(/(\$[1-9])|(([^$]|\$([^1-9]|$))+)/g) ||
[]).map(function(p) {
// Pieces that are not $1-9 should be returned after replacing $$ // with $.
if (!p.match(/^\$[1-9]$/)) {
expect(
(p.match(/\$/g) || []).length % 2 === 0, 'Unescaped $ found in localized string.');
return {value: p.replace(/\$\$/g, '$'), arg: null}; }
```

```
// Otherwise, return the substitution value.
return {value: varArgs[p[1]], arg: p}; });
return pieces; }
/**
 * As above, but also makes sure that the value is a boolean.
 * @param {string} id The key that identifies the desired boolean. * @return {boolean}
   The corresponding boolean value.
 */
getBoolean(id) {
const value = this.getValue(id); expectIsType(id, value, 'boolean'); return /** @type
{boolean} */ (value);
}
/**
 * As above, but also makes sure that the value is an integer.
 * @param {string} id The key that identifies the desired number. * @return {number} The
   corresponding number value.
 */
getInteger(id) {
const value = this.getValue(id);
expectIsType(id, value, 'number');
expect(value === Math.floor(value), 'Number isn\'t integer: ' + value); return /** @type
{number} */ (value);
}
/**
 * Override values in loadTimeData with the values found in |replacements|. * @param
   {Object} replacements The dictionary object of keys to replace. */
overrideValues(replacements) { expect(
typeof replacements === 'object',
'Replacements must be a dictionary object.'); for (const key in replacements) {
this.data_[key] = replacements[key]; }
}
/**
 * Reset loadTimeData's data. Should only be used in tests.
 * @param {?Object} newData The data to restore to, when null restores to * unset state.
 */
resetForTesting(newData = null) { this.data_ = newData;
}
/**
 * @return {boolean} Whether loadTimeData.data has been set. */
isInitialized() {
return this.data_ !== null;
} }
/**
 * Checks condition, throws error message if expectation fails.
 * @param {*} condition The condition to check for truthiness.
 * @param {string} message The message to display if the check fails. */
```

```
function expect(condition, message) { if (!condition) {
throw new Error(
'Unexpected condition on ' + document.location.href + ': ' + message);
} }
/**
* Checks that the given value has the given type.
* @param {string} id The id of the value (only used for error message). * @param {*}
value The value to check the type on.
* @param {string} type The type we expect |value| to be.
*/
function expectIsType(id, value, type) { expect(
typeof value === type, '[' + value + '] (' + id + ') is not a ' + type); }
expect(!loadTimeData, 'should only include this file once'); loadTimeData = new
LoadTimeData;
// Expose |loadTimeData| directly on |window|, since within a JS module the // scope is
local and not all files have been updated to import the exported // |loadTimeData|
explicitly.
window.loadTimeData = loadTimeData;
console.warn('crbug/1173575, non-JS module files deprecated.');
```