**Compliance**

**Video transcript**

–

For every industry, there are specific standards that need to be upheld, and you will be audited or inspected to ensure that you have met those standards. For example, for a coffee shop, the health inspector will come by and check that everything is up to code and sanitary. Similarly, you could be audited for taxes to see that you have run the back office correctly and have followed the law. You rely on documentation, records and inspections to pass audits and compliance checks as they come along.

You'll need to devise a similar way to meet compliance and auditing in AWS. Depending on what types of solutions you host on AWS, you will need to ensure that you are up to compliance for whatever standards and regulations your business is specifically held to. If you run software that deals with consumer data in the EU, you would need to make sure that you're in compliance with GDPR, or if you run healthcare applications in the US you will need to design your architectures to meet HIPAA compliance requirements.

Whatever your compliance need is, you'll need some tools to be able to collect documents, records and inspect your AWS environment to check if you meet the compliance regulations that you're under. The first thing to note is, AWS has already built out data center infrastructure and networking following industry best practices for security, and as an AWS customer, you inherit all the best practices of AWS policies, architecture, and operational processes.

AWS complies with a long list of assurance programs that you can find online. This means that segments of your compliance have already been completed, and you can focus on meeting compliance within your own architectures that you build on top of AWS. The next thing to know in regards to compliance and AWS, is that the Region you choose to operate out of, might help you meet compliance regulations. If you can only legally store data in the country that the data is from, you can choose a Region that makes sense for you and AWS will not automatically replicate data across Regions.

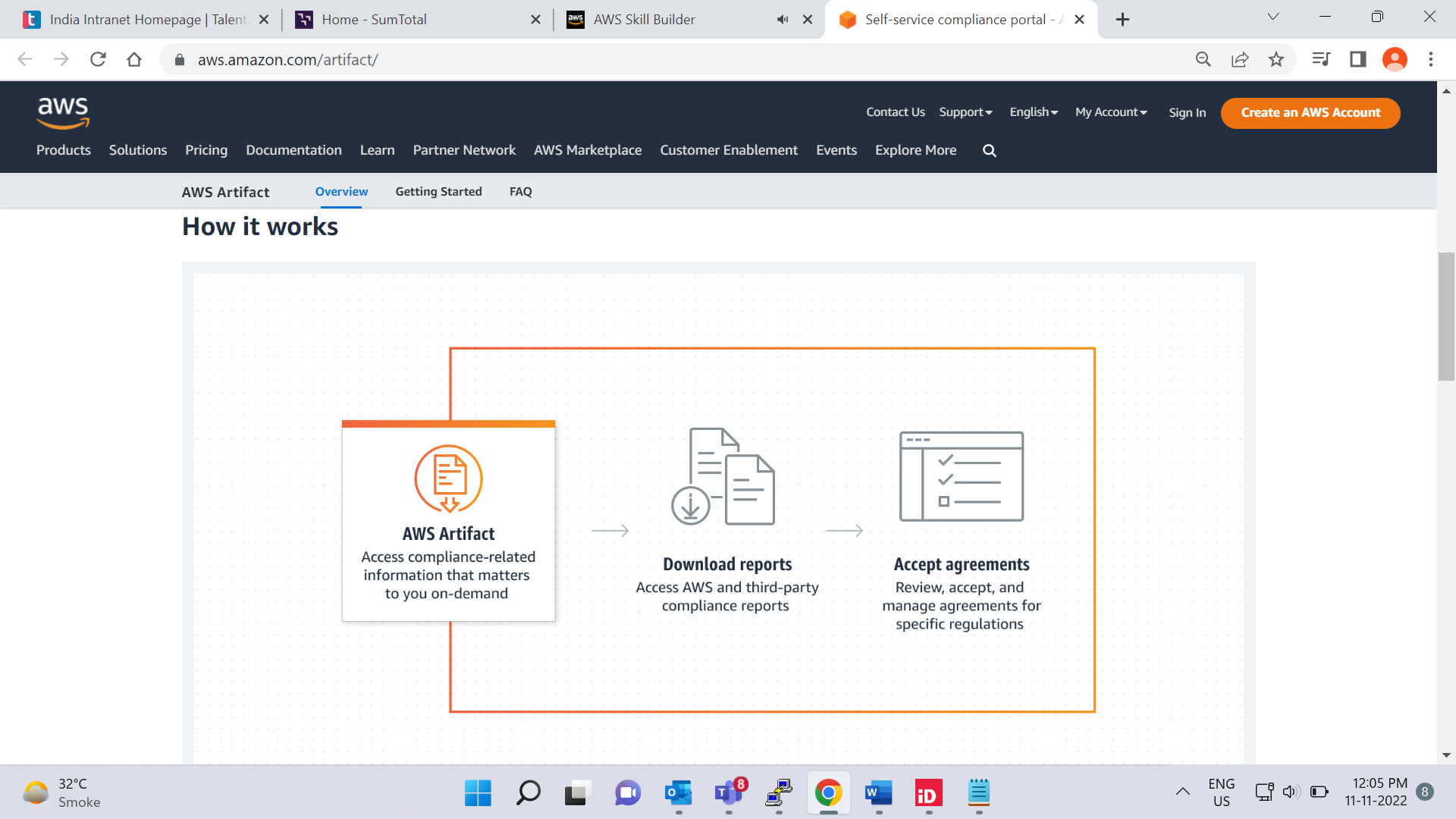
You also should be very aware of the fact that you own your data in AWS. As shown in the AWS shared responsibility model, you have complete control over the data that you store in AWS. You can employ multiple different encryption mechanisms to keep your data safe, and that varies from service to service. So, if you need specific standards for data storage, you can devise a way to either reach those requirements by building it yourself on top of AWS or using the features that already exist in many services. For a lot of services, enabling data protection is a configuration setting on the resource.

AWS also offers multiple whitepapers and documents that you can download and use for compliance reports. Since you aren't running the data center yourself, you can essentially request that AWS provides you with documentation proving that they are following best practices for security and compliance.

One place you can access these documents is through a service called AWS Artifact. With AWS Artifact, you can gain access to compliance reports done by third parties who have validated a wide range of compliance standards. Check out the AWS Compliance Center in order to find compliance information all in one place. It will show you compliance enabling services as well as documentation like the AWS Risk and Security Whitepaper, which you should read to ensure that you understand security and compliance with AWS.

To know if you are compliant in AWS, please remember that we follow a shared responsibility. The underlying platform is secure and AWS can provide documentation on what types of compliance requirements they meet, through services like AWS Artifact and whitepapers. But, beyond that, what you build on AWS is up to you. You control the architecture of your applications and the solutions you build, and they need to be built with compliance, security, and the shared responsibility model in mind.

**AWS Artifact**



Depending on your company’s industry, you may need to uphold specific standards. An audit or inspection will ensure that the company has met those standards.

[**AWS Artifact**](https://aws.amazon.com/artifact) is a service that provides on-demand access to AWS security and compliance reports and select online agreements. AWS Artifact consists of two main sections: AWS Artifact Agreements and AWS Artifact Reports.

To learn more, select the **+** symbol next to each section.

**AWS Artifact Agreements**

–

Suppose that your company needs to sign an agreement with AWS regarding your use of certain types of information throughout AWS services. You can do this through **AWS Artifact Agreements**.

In AWS Artifact Agreements, you can review, accept, and manage agreements for an individual account and for all your accounts in AWS Organizations. Different types of agreements are offered to address the needs of customers who are subject to specific regulations, such as the Health Insurance Portability and Accountability Act (HIPAA).

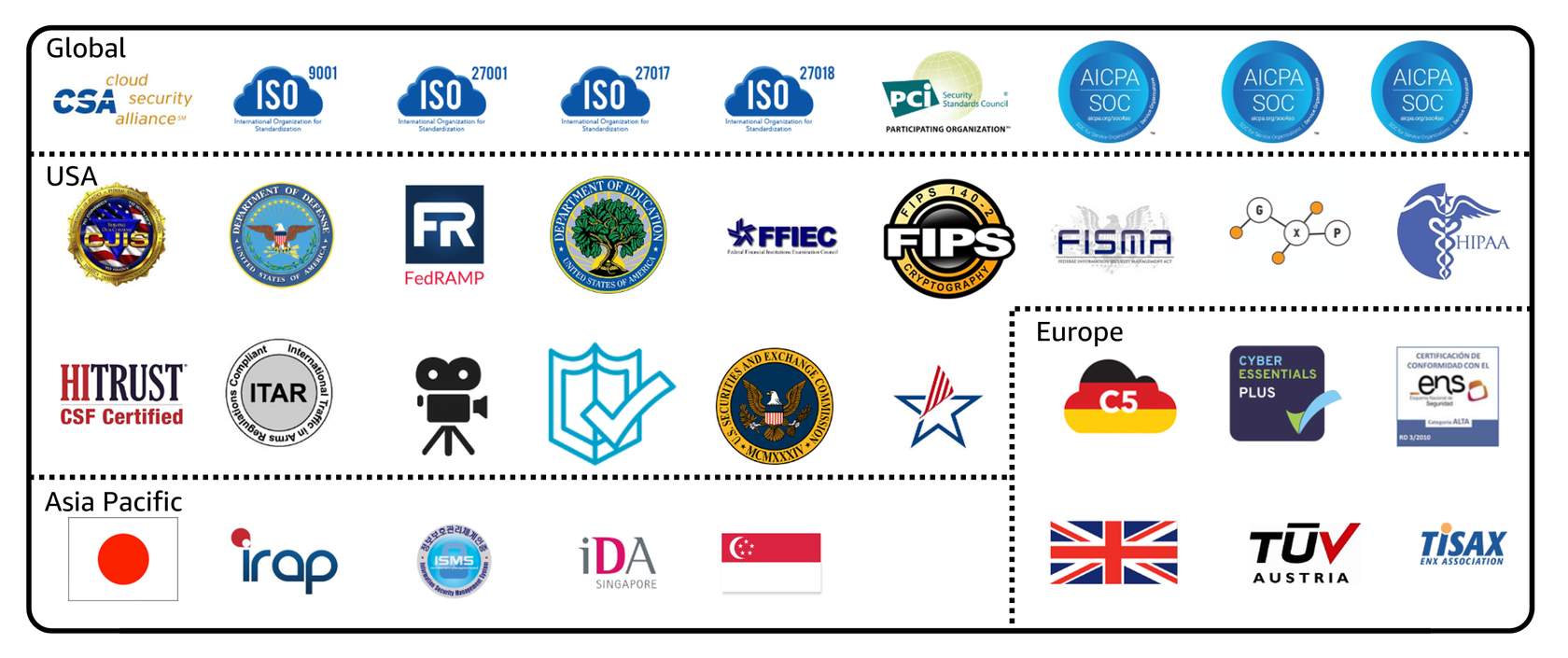
**AWS Artifact Reports**

–

Next, suppose that a member of your company’s development team is building an application and needs more information about their responsibility for complying with certain regulatory standards. You can advise them to access this information in **AWS Artifact Reports**.

AWS Artifact Reports provide compliance reports from third-party auditors. These auditors have tested and verified that AWS is compliant with a variety of global, regional, and industry-specific security standards and regulations. AWS Artifact Reports remains up to date with the latest reports released. You can provide the AWS audit artifacts to your auditors or regulators as evidence of AWS security controls.

The following are some of the compliance reports and regulations that you can find within AWS Artifact. Each report includes a description of its contents and the reporting period for which the document is valid.



**Customer Compliance Center**

The [**Customer Compliance Center**](https://aws.amazon.com/compliance/customer-center/) contains resources to help you learn more about AWS compliance.

In the Customer Compliance Center, you can read customer compliance stories to discover how companies in regulated industries have solved various compliance, governance, and audit challenges.

You can also access compliance whitepapers and documentation on topics such as:

* AWS answers to key compliance questions
* An overview of AWS risk and compliance
* An auditing security checklist

Additionally, the Customer Compliance Center includes an auditor learning path. This learning path is designed for individuals in auditing, compliance, and legal roles who want to learn more about how their internal operations can demonstrate compliance using the AWS Cloud.

**Knowledge check**

Which tasks can you complete in AWS Artifact? (Select TWO.)

* Access AWS compliance reports on-demand.

Correctly checked

* Consolidate and manage multiple AWS accounts within a central location.

Correctly unchecked

* Create users to enable people and applications to interact with AWS services and resources.

Correctly unchecked

* Set permissions for accounts by configuring service control policies (SCPs).

Correctly unchecked

* Review, accept, and manage agreements with AWS.

Correctly checked

SUBMIT

**Correct**

The correct two response options are:

* **Access AWS compliance reports on-demand.**
* **Review, accept, and manage agreements with AWS.**

The other response options are incorrect because:

* Consolidate and manage multiple AWS accounts within a central location- This task can be completed in *AWS Organizations*.
* Create users to enable people and applications to interact with AWS services and resources- This task can be completed in *AWS Identity and Access Management (IAM)*.
* Set permissions for accounts by configuring service control policies (SCPs)- This task can be completed in *AWS Organizations*.

# Denial-of-service attacks

**Video transcript**

–

D-D-o-S, DDoS, the distributed denial-of-service. It's an attack on your enterprise's infrastructure, and you've heard of it. Your security team might have written a plan for it, and you know that many businesses have been devastated by it. But what exactly is it, and more importantly, how can you defend against it?

Now to be clear, this is a 14-hour discussion to really understand it all, but you need to at least know the fundamentals of how the attacks are carried out, and how AWS can automatically defend your infrastructure from these crippling assaults. Now we don't have a lot of time to cover all this, and the clock starts now. The objective of a DDoS attack is to shut down your application's ability to function by overwhelming the system to the point it can no longer operate.

In normal operations, your application takes requests from customers and returns results. In a DDoS attack, the bad actor tries to overwhelm the capacity of your application, basically to deny anyone your services. But a single machine attacking your application has no hope of providing enough of an attack by itself, so the distributed part is that the attack leverages other machines around the internet to unknowingly attack your infrastructure. The bad actor creates an army of zombie bots, brainlessly assaulting your enterprise. The key to a good attack, and I call it that when I should call it powerful. I mean, it's definitely chaotic evil, but the key is to have the assault commander do the smallest amount of work needed, and have the targeted victim receive an unbearable load of resulting work they must process through.

So let me cherry-pick a few specific attack examples that work really well. The UDP flood. It is based on the helpful parts of the internet, like the National Weather Service. Now anyone can send a small request to the Weather Service, and ask, "Give me weather," and in return, the Weather Service's fleet of machines will send back a massive amount of weather telemetry, forecasts, updates, lots of stuff. So the attack here is simple. The bad actor sends a simple request, give me weather. But it gives a fake return address on the request, your return address. So now the Weather Service very happily floods your server with megabytes of rain forecasts, and your system could be brought to a standstill, just sorting through the information it never wanted in the first place. Now that is one example of half a dozen low-level, brute force attacks, all designed to exhaust your network.

Some attacks are much more sophisticated, like the HTTP level attacks, which look like normal customers asking for normal things like complicated product searches over and over and over, all coming from an army of zombified bot machines. They ask for so much attention that regular customers can't get in.

They even try horrible tricks like the Slowloris attack. Mm-hmm. Imagine standing in line at the coffee shop, when someone in front of you takes seven minutes to order their whatever it is they're ordering, and you don't get to order until they finish and get out of your way. Well, Slowloris attack is the exact same thing. Instead of a normal connection, I would like to place an order, the attacker pretends to have a terribly slow connection. You get the picture. Meanwhile, your production servers are standing there waiting for the customer to finish their request so they can dash off and return the result. But until they get the entire packet, they can't move on to the next thread, the next customer. A few Slowloris attackers can exhaust the capacity of your entire front end with almost no effort at all. I could go on monologuing for hours just talking about the elegantly evil architecture of these attacks, but we are on the clock here, and it is time to stop these attacks cold. And here's the cool solution: You already know the solution.

Everything we've been talking about over this entire course is not only good architecture, but it also helps solve almost all DDoS attack vectors with zero additional effort or cost. First attack, the low level network attacks like the UDP floods. Solution, security groups. The security groups only allow in proper request traffic. Things like weather reports use an entirely different protocol than the ones your customers use. Not on the list, you don't get to talk to the server. And what's more, security groups operate at the AWS network level, not at the EC2 instance level, like an operating system firewall might.

So massive attacks like UDP floods or reflection attacks just get shrugged off by the scale of the entire AWS Regions capacity, not your individual EC2's capacity. This is a case where our size is a huge advantage in your protection. I won't say it's impossible to overwhelm AWS, but the scale it would take, it would be too expensive for these bad actors. Slowloris attacks? Look at our elastic load balancer. Because the ELB handles the http traffic request first, so it waits until the entire message, no matter how fast or slow, is complete before sending it over to the front end web server. I mean, sure, you can try to overwhelm it, but remember how the ELB is scalable and how it runs at the region level?

To overwhelm ELB, you would once again have to overwhelm the entire AWS region. It's not theoretically impossible, but too massively expensive for anyone to pull off. For the sharpest, most sophisticated attacks, AWS also offers specialized defense tools called AWS Shield with AWS WAF. AWS WAF uses a web application firewall to filter incoming traffic for the signatures of bad actors. It has extensive machine learning capabilities, and can recognize new threats as they evolve and proactively help defend your system against an ever-growing list of destructive vectors.

All right, that's it, the clock is almost up. The takeaway is a well-architected system is already defended against most attacks. And by using AWS Shield Advanced, you can turn AWS into your partner against DDoS attacks. Oh, oh.

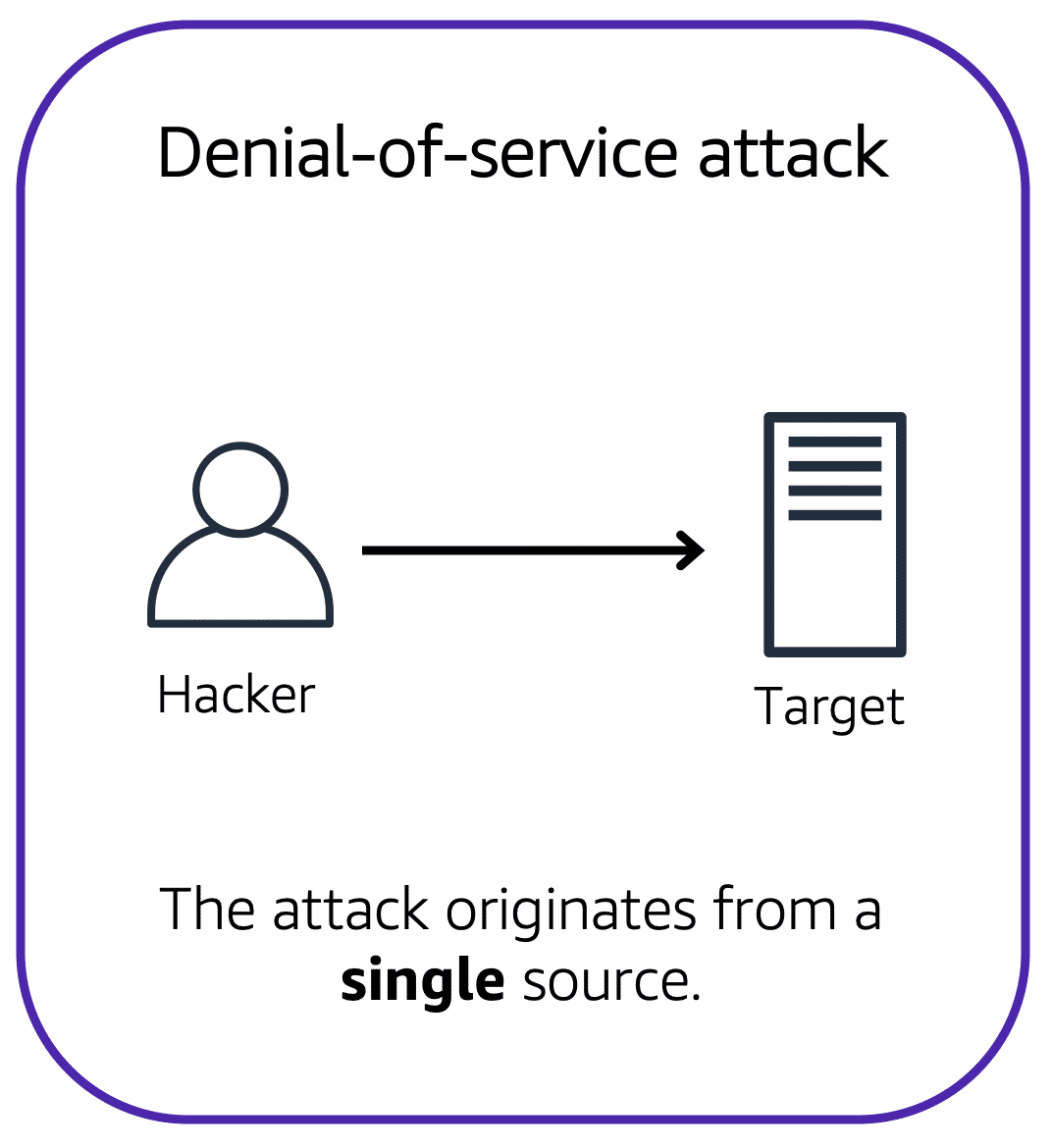
Customers can call the coffee shop to place their orders. After answering each call, a cashier takes the order and gives it to the barista.

However, suppose that a prankster is calling in multiple times to place orders but is never picking up their drinks. This causes the cashier to be unavailable to take other customers’ calls. The coffee shop can attempt to stop the false requests by blocking the phone number that the prankster is using.

In this scenario, the prankster’s actions are similar to a **denial-of-service attack**.

**Denial-of-service attacks**

A **denial-of-service (DoS) attack** is a deliberate attempt to make a website or application unavailable to users.

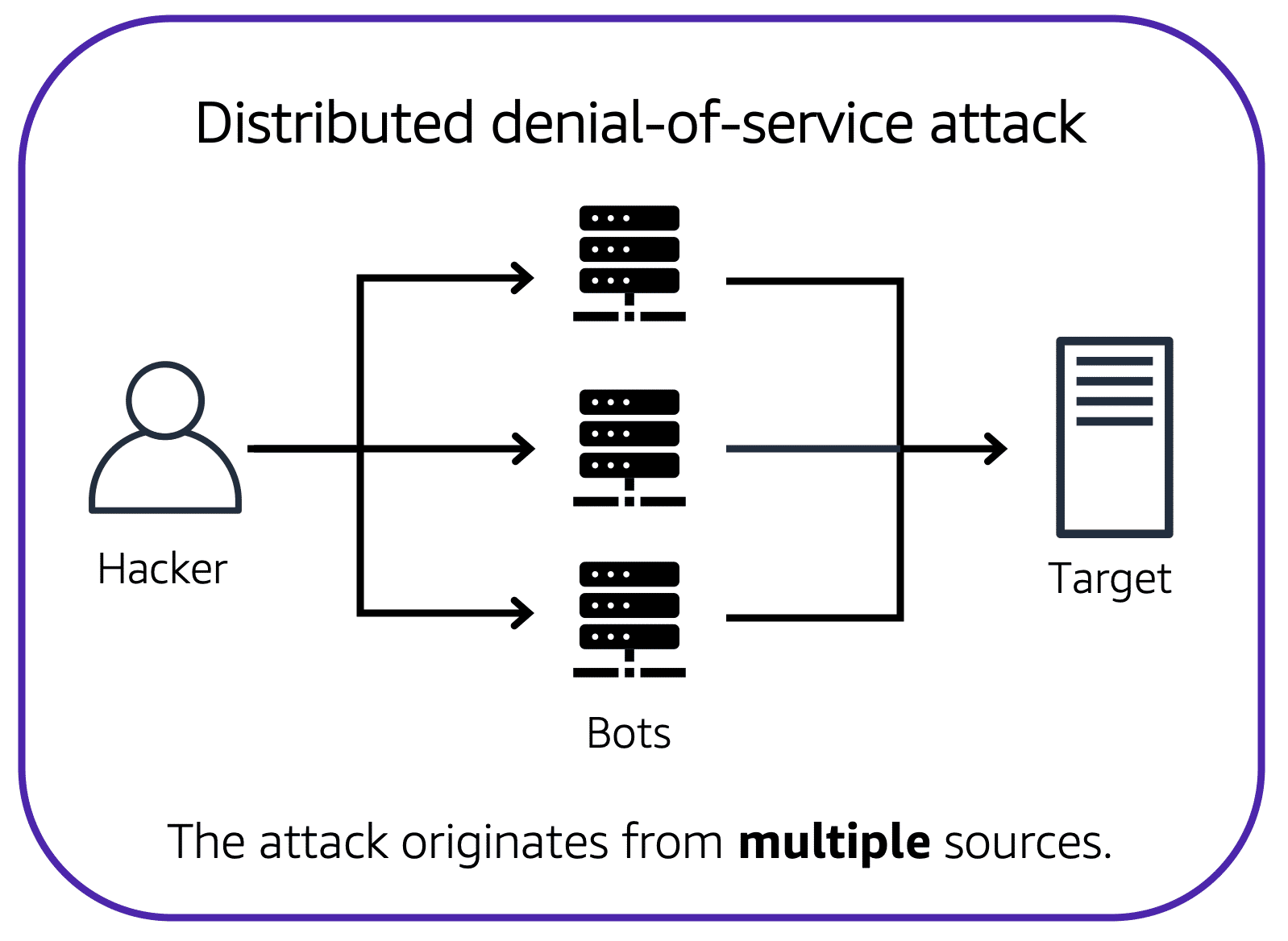


For example, an attacker might flood a website or application with excessive network traffic until the targeted website or application becomes overloaded and is no longer able to respond. If the website or application becomes unavailable, this denies service to users who are trying to make legitimate requests.

**Distributed denial-of-service attacks**

Now, suppose that the prankster has enlisted the help of friends.

The prankster and their friends repeatedly call the coffee shop with requests to place orders, even though they do not intend to pick them up. These requests are coming in from different phone numbers, and it’s impossible for the coffee shop to block them all. Additionally, the influx of calls has made it increasingly difficult for customers to be able to get their calls through. This is similar to a **distributed denial-of-service attack**.



In a distributed denial-of-service (DDoS) attack, multiple sources are used to start an attack that aims to make a website or application unavailable. This can come from a group of attackers, or even a single attacker. The single attacker can use multiple infected computers (also known as “bots”) to send excessive traffic to a website or application.

To help minimize the effect of DoS and DDoS attacks on your applications, you can use [**AWS Shield**](https://aws.amazon.com/shield).

**AWS Shield**

AWS Shield is a service that protects applications against DDoS attacks. AWS Shield provides two levels of protection: Standard and Advanced.

To learn more, select the **+** symbol next to each service.

**AWS Shield Standard**

–

**AWS Shield Standard** automatically protects all AWS customers at no cost. It protects your AWS resources from the most common, frequently occurring types of DDoS attacks.

As network traffic comes into your applications, AWS Shield Standard uses a variety of analysis techniques to detect malicious traffic in real time and automatically mitigates it.

**AWS Shield Advanced**

–

**AWS Shield Advanced** is a paid service that provides detailed attack diagnostics and the ability to detect and mitigate sophisticated DDoS attacks.

It also integrates with other services such as Amazon CloudFront, Amazon Route 53, and Elastic Load Balancing. Additionally, you can integrate AWS Shield with AWS WAF by writing custom rules to mitigate complex DDoS attacks.

# Additional security services

**Video transcript**

–

With all the comings and goings on in your coffee shop, you'll want to increase security to your coffee beans, equipment, and even money in the till. For your beans, this could be when they're sitting in your store room, or even when you're transporting them between shops. After all, we don't want unwanted visitors with access to our coffee beans, or even running off with precious equipment.

To start off, let's chat about how you can secure your coffee beans, or your data whether it's at rest, or in transit. For our beans, the simple way to do it would be to lock the door when we'd leave at night. That's the notion of encryption, which is securing a message or data in a way that can only be accessed by authorized parties. Non-authorized parties are therefore less likely to be able to access the message. Or not able to access it at all. Think of it as that key and door example. If you have the key, you can unlock the door. But if you don't, then you cannot unlock that door.

At AWS, this comes in two variations. Encryption at rest and encryption in transit. By at rest, we mean when your data is idle. It's just being stored and not moving. For example, server-side encryption at rest is enabled on all DynamoDB table data. And that helps prevent unauthorized access. DynamoDB's encryption at rest also integrates with AWS KMS, or Key Management Service, for managing the encryption key that is used to encrypt your tables. That's the key for your door, remember? And without it, you won't be able to access your data. So make sure to keep it safe.

Similarly, in-transit means that the data is traveling between, say A and B. Where A is the AWS service, and B could be a client accessing the service. Or even another AWS service itself. For example, let's say we have a Redshift instance running. And we want to connect it with a SQL client. We use secure sockets layer, or SSL connections to encrypt data, and we can use service certificates to validate, and authorize a client. This means that data is protected when passing between Redshift, and our client. And this functionality exists in numerous other AWS services such as SQS, S3, RDS, and many more.

But speaking of other services, the next service we want to highlight is called Amazon Inspector. Inspector helps to improve security, and compliance of your AWS deployed applications by running an automated security assessment against your infrastructure. Specifically, it helps to check on deviations of security best practices, exposure of EC2 instances, vulnerabilities, and so forth. The service consists of three parts a network configuration reachability piece, an Amazon agent, which can be installed an EC2 instances, and a security assessment service that brings them all together. To use it, you configure Inspector options, run the service, out pops a list of potential security issues. The resulting findings are displayed in the Amazon Inspector console, and they are presented with a detailed description of the security issue, and a recommendation on how to fix it. Additionally, you can retrieve findings through an API. So as to go towards the best practice of performing remediation to fix issues.

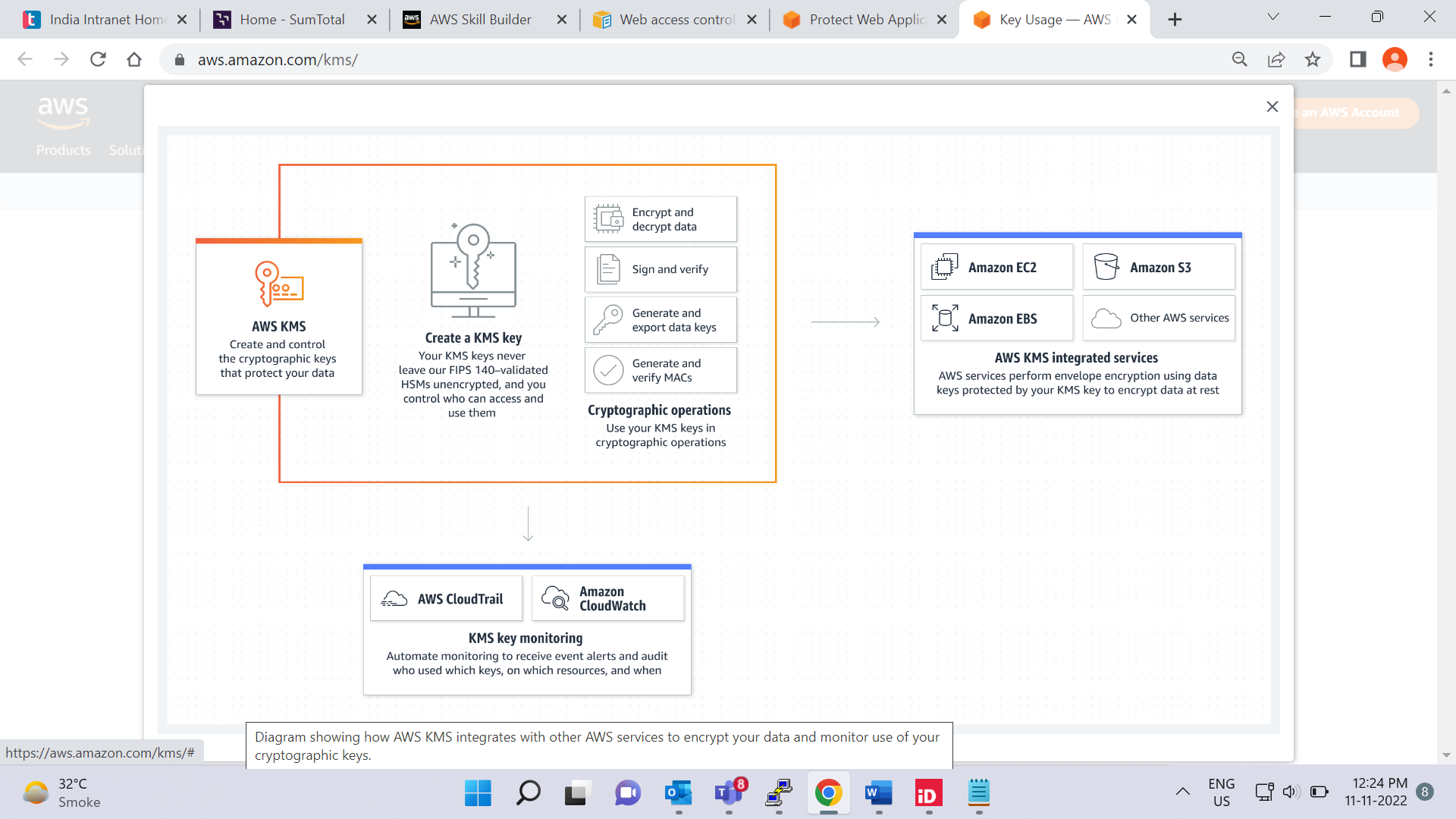
Another service dimension is our threat detection offering known as Amazon GuardDuty. It analyzes continuous streams of metadata generated from your account, and network activity found on AWS CloudTrail events, Amazon VPC Flow Logs, and DNS logs. It uses integrated threat intelligence such as known malicious IP addresses, anomaly detection, and machine learning to identify threats more accurately. The best part is that it runs independently from your other AWS services. So it won't affect performance or availability of your existing infrastructure, and workloads.

They are so many other security services like Advanced Shield and Security Hub. So please check out the Resources section to learn more. Thanks for following along.

**AWS Key Management Service (AWS KMS)**

The coffee shop has many items, such as coffee machines, pastries, money in the cash registers, and so on. You can think of these items as data. The coffee shop owners want to ensure that all of these items are secure, whether they’re sitting in the storage room or being transported between shop locations.

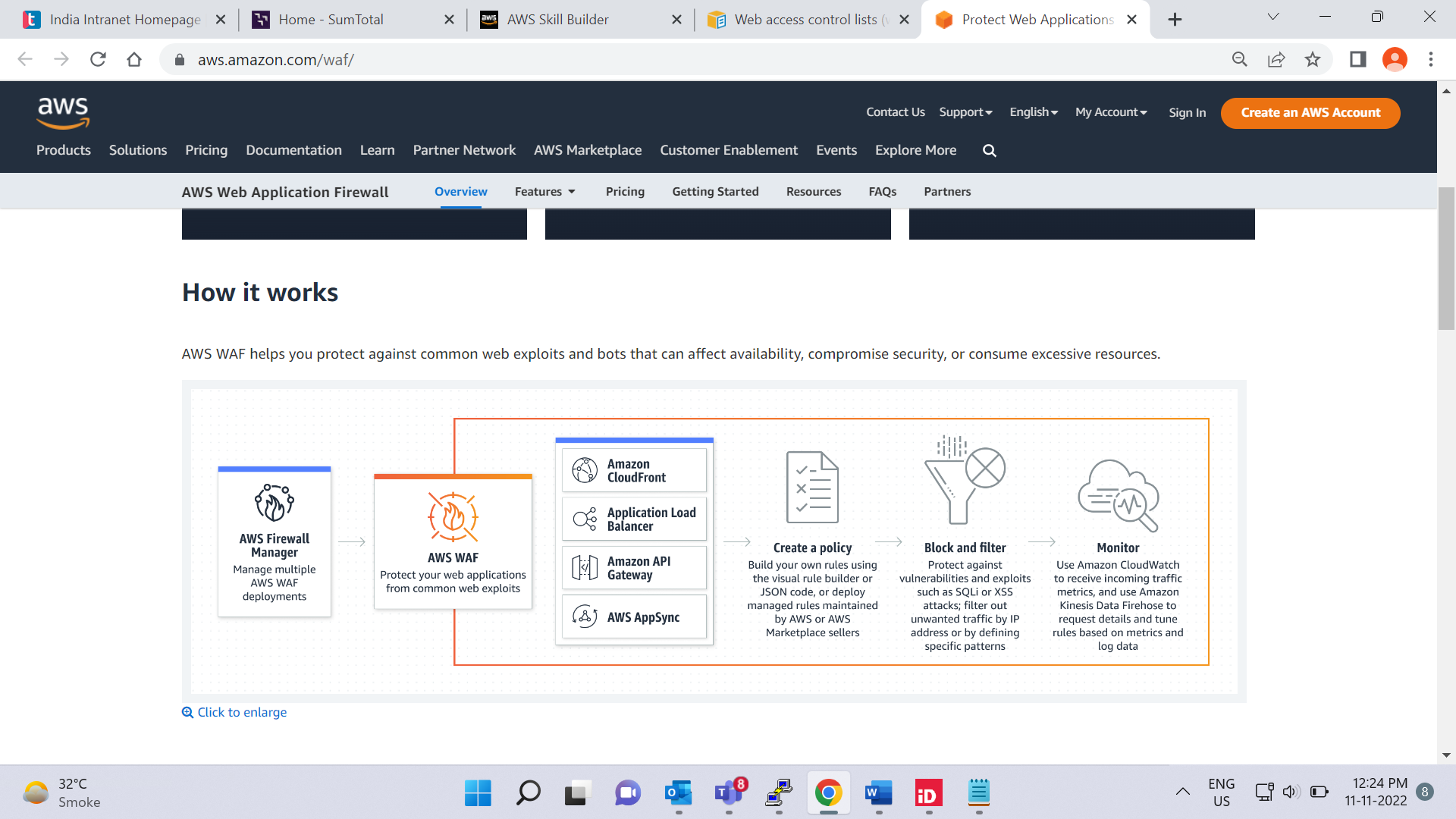
In the same way, you must ensure that your applications’ data is secure while in storage **(encryption at rest)** and while it is transmitted, known as **encryption in transit**.



[**AWS Key Management Service (AWS KMS)**](https://aws.amazon.com/kms) enables you to perform encryption operations through the use of **cryptographic keys**. A cryptographic key is a random string of digits used for locking (encrypting) and unlocking (decrypting) data. You can use AWS KMS to create, manage, and use cryptographic keys. You can also control the use of keys across a wide range of services and in your applications.

With AWS KMS, you can choose the specific levels of access control that you need for your keys. For example, you can specify which IAM users and roles are able to manage keys. Alternatively, you can temporarily disable keys so that they are no longer in use by anyone. Your keys never leave AWS KMS, and you are always in control of them.

**AWS WAF**



[**AWS WAF**](https://aws.amazon.com/waf) is a web application firewall that lets you monitor network requests that come into your web applications.

AWS WAF works together with Amazon CloudFront and an Application Load Balancer. Recall the network access control lists that you learned about in an earlier module. AWS WAF works in a similar way to block or allow traffic. However, it does this by using a [**web access control list (ACL)**](https://docs.aws.amazon.com/waf/latest/developerguide/web-acl.html) to protect your AWS resources.

Here’s an example of how you can use AWS WAF to allow and block specific requests.

# Module 6 summary

In Module 6, you learned about the following concepts:

* The shared responsibility model
* Features of AWS Identity and Access Management
* Methods of managing multiple accounts in AWS Organizations
* AWS compliance resources
* AWS services for application security and encryption

**Video transcript**

–

Alright, it's time to break down what we just covered. First things first, AWS follows a shared responsibility model. AWS is responsible for security of the cloud, and you are responsible for security in the cloud.

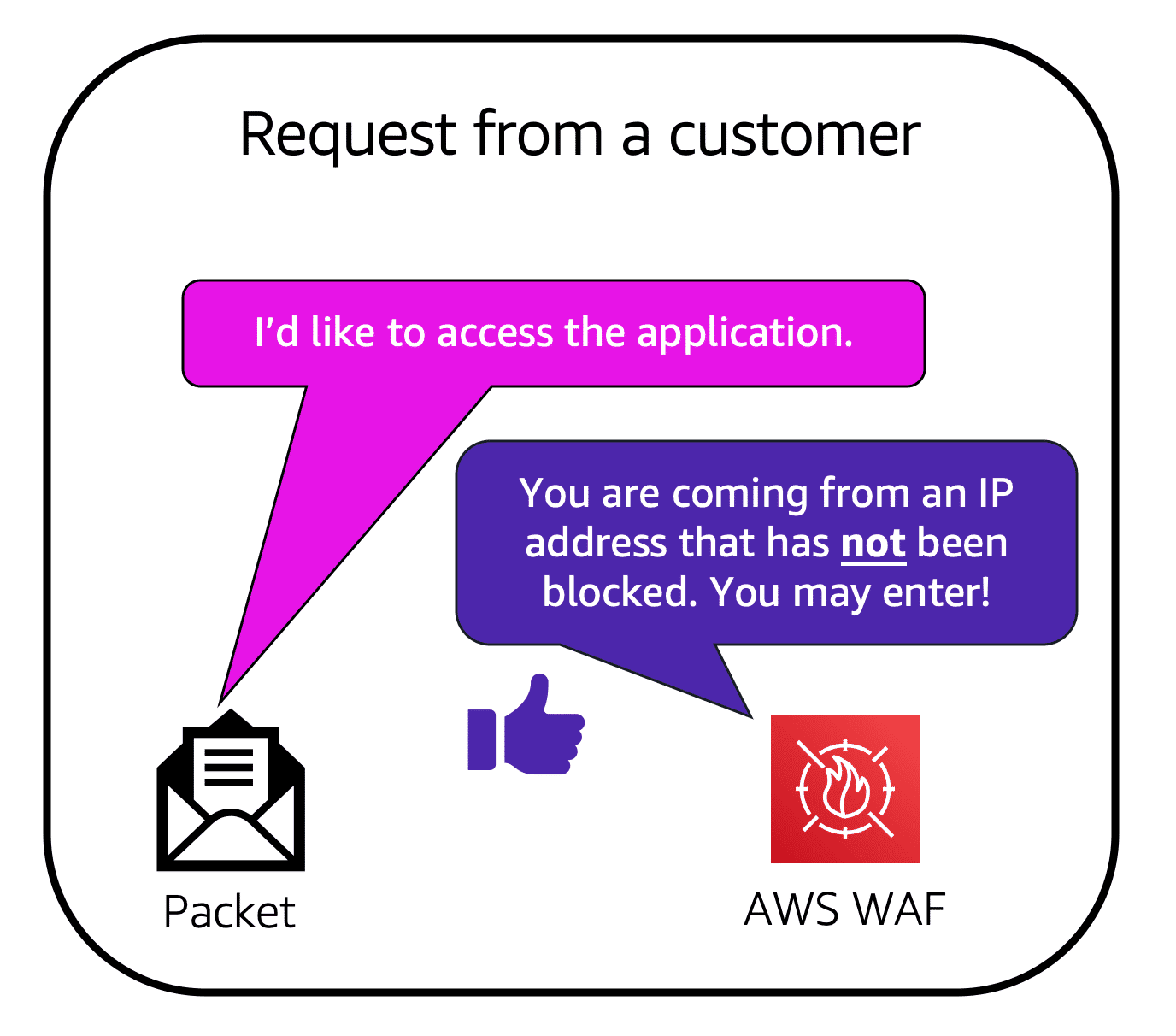
With IAM, you have users, groups, roles, and policies. Users log in with a username and password and by default they have no permissions. Groups are groupings of users and roles are identities that you can assume to gain access to temporary credentials and permissions for a configurable amount of time. In order to give permissions to an identity, you need to create policies that either explicitly allow or deny a specific action in AWS. With IAM also comes identity federation. If you have an existing corporate identity store, you can federate those users to AWS, using role based access, which allows your users to use one login for both your corporate systems as well as AWS. One final point to remember about IAM is that you should make sure that you turn on multi-factor authentication for users, but especially for your root user which has all the permissions by default and cannot be restricted.

Next up, we discussed AWS Organizations. With AWS, it's likely you'll have multiple accounts. Accounts are commonly used to isolate workloads, environments, teams, or applications. AWS Organizations helps you manage multiple accounts in a hierarchical fashion. We then discussed compliance. AWS uses third-party auditors to prove its adherence to a wide variety of compliance programs. You can use the AWS Compliance Center to find more information on compliance and AWS Artifact to gain access to compliance documents. The compliance requirements you have will vary from application to application and between areas of operation.

Then we talked about distributed denial-of-service attacks, or DDoS attacks, and how to combat them with AWS using tools like ELB, security groups, AWS Shield, and AWS WAF.

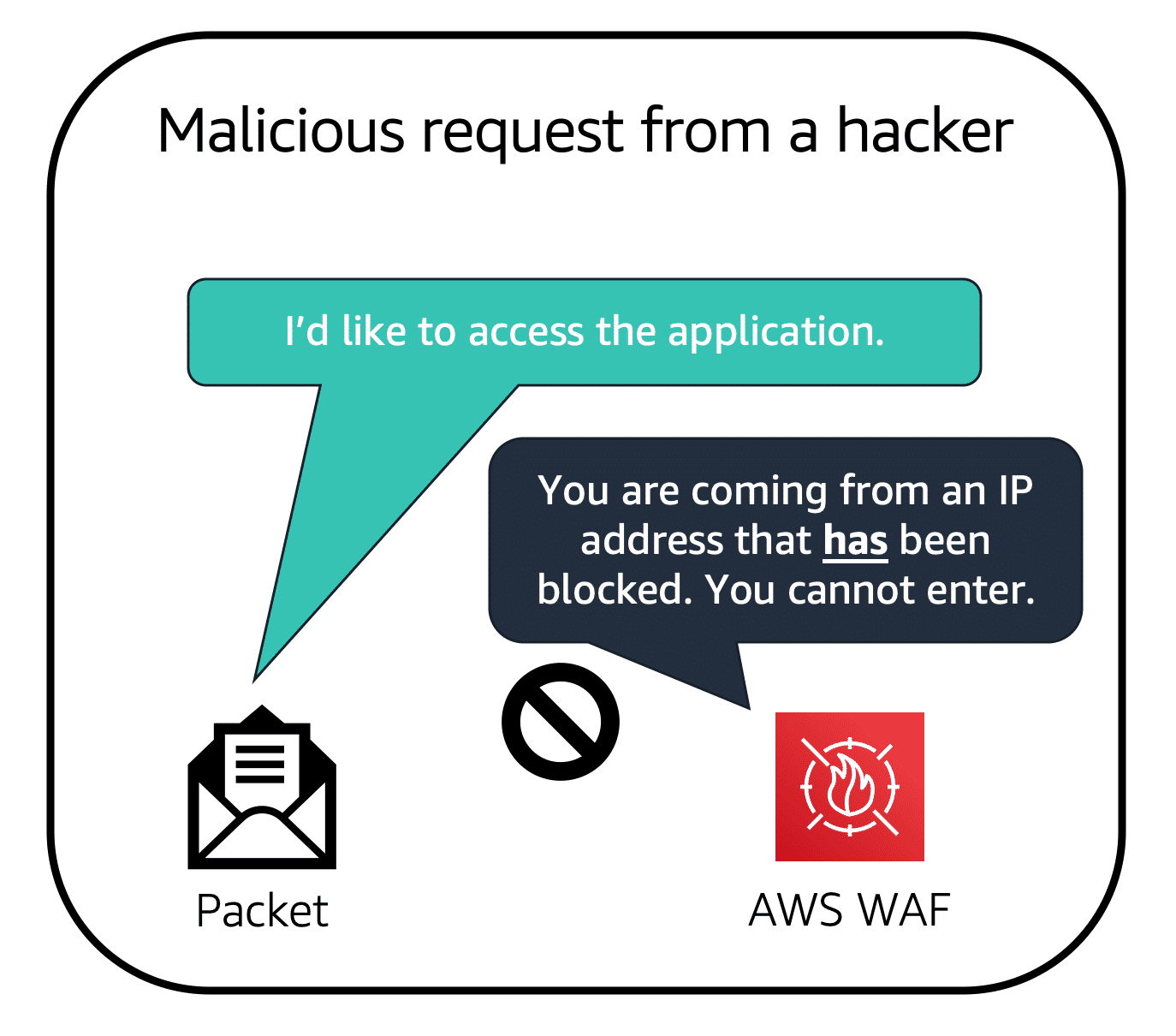
We also talked about encryption. In AWS, you are the owner of your data, and you are responsible for security. That means you need to pay attention to encryption, in transit and at rest.

There are lots of considerations when dealing with security in AWS. Security is AWS's top priority, and will continue to be so. Please make sure you read the documentation on securing your AWS resources, as it does vary from service to service. Use the least privilege principle when scoping permissions for users and roles in IAM, encrypt your data at every layer, both in transit and at rest. And make sure you use AWS services to protect your environment.



Suppose that your application has been receiving malicious network requests from several IP addresses. You want to prevent these requests from continuing to access your application, but you also want to ensure that legitimate users can still access it. You configure the web ACL to allow all requests except those from the IP addresses that you have specified.

When a request comes into AWS WAF, it checks against the list of rules that you have configured in the web ACL. If a request did not come from one of the blocked IP addresses, it allows access to the application.



However, if a request came from one of the blocked IP addresses that you have specified in the web ACL, it is denied access.

**Amazon Inspector**

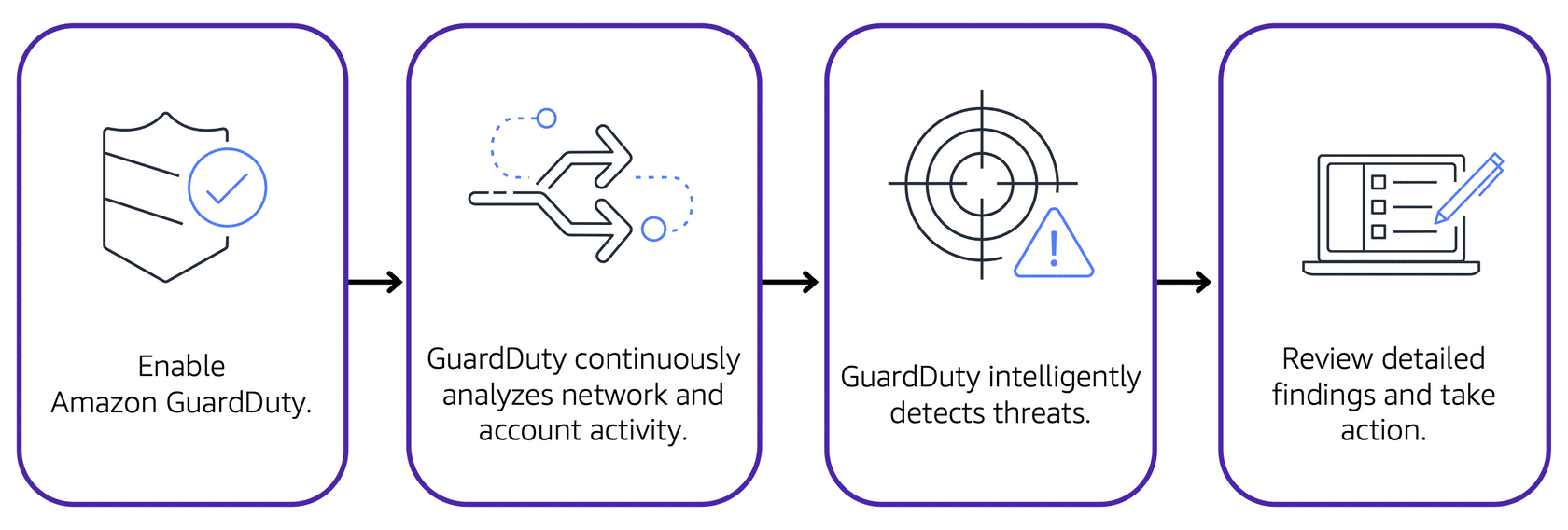
Suppose that the developers at the coffee shop are developing and testing a new ordering application. They want to make sure that they are designing the application in accordance with security best practices. However, they have several other applications to develop, so they cannot spend much time conducting manual assessments. To perform automated security assessments, they decide to use [**Amazon Inspector**](https://aws.amazon.com/inspector/).

Amazon Inspector helps to improve the security and compliance of applications by running automated security assessments. It checks applications for security vulnerabilities and deviations from security best practices, such as open access to Amazon EC2 instances and installations of vulnerable software versions.

After Amazon Inspector has performed an assessment, it provides you with a list of security findings. The list prioritizes by severity level, including a detailed description of each security issue and a recommendation for how to fix it. However, AWS does not guarantee that following the provided recommendations resolves every potential security issue. Under the shared responsibility model, customers are responsible for the security of their applications, processes, and tools that run on AWS services.

**Amazon GuardDuty**

[**Amazon GuardDuty**](https://aws.amazon.com/guardduty) is a service that provides intelligent threat detection for your AWS infrastructure and resources. It identifies threats by continuously monitoring the network activity and account behavior within your AWS environment.



After you have enabled GuardDuty for your AWS account, GuardDuty begins monitoring your network and account activity. You do not have to deploy or manage any additional security software. GuardDuty then continuously analyzes data from multiple AWS sources, including VPC Flow Logs and DNS logs.

If GuardDuty detects any threats, you can review detailed findings about them from the AWS Management Console. Findings include recommended steps for remediation. You can also configure AWS Lambda functions to take remediation steps automatically in response to GuardDuty’s security findings