Welcome to the web connectivity and

security course. In this course we will study different web

connectivity standards, solutions, and protocols. We will also study the need for

security as well as protocols, methodologies and solutions that will

help you mitigate security risks. Before talking about web connectivity

in cyber-physical systems, the first question that arises is, why then bother

connecting your embedded systems? If there is a device that controls

the settings on my microwave oven, what need does it have to

talk to other devices? This is a valid point. Standalone embedded systems do exist and cater to a specific problem as

in the case of a microwave oven. Another example is an image

processor in a digital camera. Whose purpose is handling the light

capture from the sensor and to digitize, process, and

store that information as an image. It performs that specific job and

does it very well. The reality is that devices

are getting more and more complex and that connection is needed. The same digital camera now has wi-fi and

GPS modules to offer seamless transfer of photos and

a geopack then. These wi-fi and GPS modules have

their own baseline processors and for all of them to work seamlessly,

you have to bring them together. You have to connect them. Any reasonably modern car can

have 25 to 50 onboard processors. A modern 7-Series BMW has

up to a hundred processors. All of these control different

functions of the car. For example GPS navigation, entertainment, parking distance control,

radar cruise control, and so on. To make the car really function and provides a safer journey, the processors

have to communicate with each other. A classic example is that of a car crash. In case of a crash, your

car will open

the door locks and deploy the air bag. It may perform other emergency

functions like flashing the headlights ,and sending your location

coordinates to an emergency service. All of this is possible

because of the in-car network. If you think this is too much,

NVIDIA announced car computation platform that has a combined

performance of 8 teraflops. To put this in perspective,

the world's 418th-ranked super computer in 2008 had

a computational power of 9.1 teraflops and

that one had 4,096 processor cores. So, soon your car will have

the same computation or capability as the super computers in 2008. And here we only discuss the

self-contained local networks within cars. Soon cars will be able to communicate

with other cars on the road and with the traffic monitoring systems. Thus, the future is connected and connectivity is not optional anymore,

it's fundamental. After establishing the case for

connectivity in cyber-physical or embedded systems, the next question would

be why do we need web connectivity? Here we need to have a deeper look. Let's first look into

the benefits of web connectivity.

**The first and foremost benefit is ubiquity**. The web is everywhere, from your PC,

to phones, to even smart watches. You can access the world wide

web not only on smartphones but also on dumb phones from a decade ago. All you need is a browser. A browser like Opera Mini that can

even be bundled into embedded systems.

**Scalability is another important point.** The World Wide Web is already quite large and it can accommodate even more

sources of information and users. Web technologies can automatically scale, based on demand,

with minimal user intervention. They can scale to user needs,

computational load, or network requirements. The World Wide Web has been with us for

well more than two decades now. That is enough time to test and

optimize the web-enabling technologies, HTML, http, and

web servers are fully understood. There are highly optimized web servers available for small sensors that take less than seven kilobytes of space, which is an efficient use of memory. One of the main features of web technologies is that they don't care about the underlying architecture or technologies. One web-supported device can communicate effectively with another one, irrespective of the system and communication infrastructure used.

**Lastly, major future technologies will reside on the web or heavily utilize the current internet technologies.** Let's take cloud computing where you are provided with on-demand computational power and services. You can use these services for big data analytics and then utilize these data insights for better decision-making.

**Imagine gathering data from hundreds of thousands of sensors, then transferring this data to the cloud for storage and processing, and afterwards relaying this decision back to the nodes or an actuator to activate.**

Despite the benefits of web connectivity, it also comes at a cost. With web connectivity, you get to leverage the cloud computing infrastructure, remote monitoring and assessment as well as remote command and control. However, the very first thing for people involved with design and development of connected embedded systems is an increase in complexity. Compared to a stand alone setup, a network connected system requires extra effort In network design and connectivity solutions, and in hardware and software design to enable that connectivity. In later videos we'll discuss more on the choices you have when it comes to network connectivity.

**Moreover, with web connectivity you inherit its security issues as well.** When you connect your cyber-physical systems through the internet, you open then to roughly the same level of security weaknesses and attacks as in other Internet-connected networks. Security can't be an afterthought anymore. If you are building a connected system, security should be an inherent

feature right from the beginning. In case of the car example, you have
a network limited to the car's internals. So, locking the door
will effectively lock out most thieves from accessing
the computer systems. But when you connect
that car to the internet, you can't rely on locking the doors only. You have opened the car to attacks
from anywhere in the world.

**But be assured, we have technologies, protocols, and solutions that will help**
**mitigate these threats. And we'll discuss them in later videos. And lastly web connectivity**
is resource heavy. For really small low power devices,
memory and computational power are a limited resource, connectivity
puts an extra burden on these resources. Depending upon the connectivity solutions,
solutions that you choose, you will have to add
the extra hardware and software layers to communicate and that
takes both memory and processing power. Some solutions are more
resource hungry than others and we will discuss that further later on. But nonetheless,
connectivity adds an extra cost. Additionally, if you decide to
add security, which you should, you'll need some sort of cryptographic
algorithm to secure your data. Cryptographic computations
are processor-intensive, meaning that they put a lot
of strain on the processor. This can drive up the energy
consumption as well. Let's discuss connectivity
in cyber-physical systems in a general context. Of course, it's nice to have an overview
of your home when you are on vacation or remote start your car in a freezing
weather so, it's warm when you get inside. Or to be able to assess what's
wrong in the production plant without physically being there. But, you will come across situations
where you won't need web connectivity. For our very daily life for example, do you really need to connect your
stand-alone coffee machine to your wi-fi? Of course you might think,

it's just a coffee machine. What can go wrong if hackers get access to it? Over boil my coffee to give it a bitter taste? The thing is that through your coffee machine if its security is weak, hackers will get access to your network and from there they can go to your system or anywhere. In this case, the coffee machine is not the target itself but, another attack vector against your system. So, yes, there are benefits to web connectivity and there are the downsides. What you will need to do is assess whether you really need web connectivity or not. Once you decide on the former, you'll need a proper implementation. That is where this course will help you. By the end, you'll be more informed about the protocols and solutions available to you. And you will be able to make informed decisions when it comes to adding connectivity to your embedded solutions. [MUSIC]