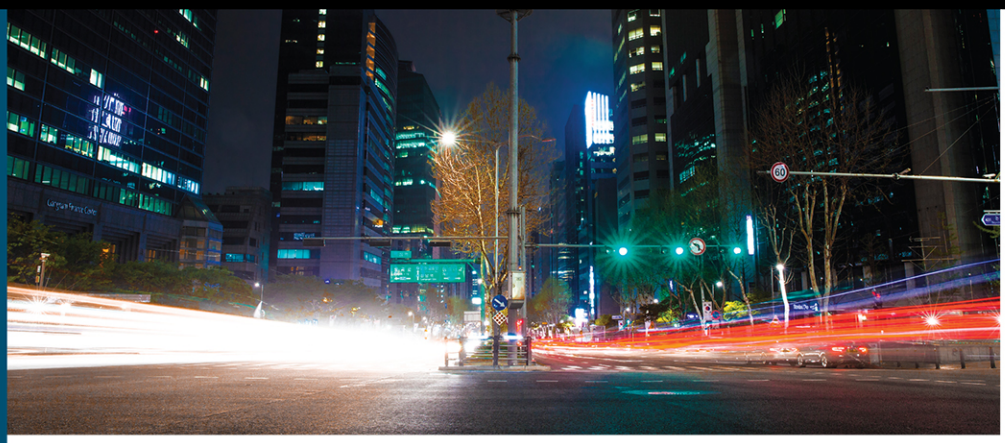


Understanding and Troubleshooting Cisco Catalyst 9800 Series Wireless Controllers

SIMONE ARENA
FRANCISCO SEDANO CRIPPA, CCIE® NO. 14859
NICOLAS DARCHIS, CCIE® NO. 25344
SUDHA KATGERI, CCIE® NO. 45857

ciscopress.com

Understanding and Troubleshooting Cisco Catalyst 9800 Series Wireless Controllers

Simone Arena
Francisco Sedano Crippa, CCIE No. 14859
Nicolas Darchis, CCIE No. 25344
Sudha Katgeri, CCIE No. 45857

Cisco Press

Understanding and Troubleshooting Cisco Catalyst 9800 Series Wireless Controllers

Simone Arena
Francisco Sedano Crippa, CCIE #14859
Nicolas Darchis, CCIE #25344
Sudha Katgeri, CCIE # 45857

Published by:
Cisco Press

ScoutAutomatedPrintCode

Library of Congress Control Number: 2022906015

ISBN-13: 978-0-13-749232-9

ISBN-10: 0-13-749232-4

Warning and Disclaimer

This book is about deploying and troubleshooting a wireless network with the next generation Catalyst 9800 Wireless Controller. It covers the software and hardware architecture, the design and deployment aspects and provides useful troubleshooting tools. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an “as is” basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the authors and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Editor-in-Chief: Mark Taub

Alliances Manager, Cisco Press: Arezou Gol

Director, ITP Product Management: Brett Bartow

Executive Editor: Nancy Davis

Managing Editor: Sandra Schroeder

Development Editor: Ellie C. Bru

Project Editor: Mandie Frank

Copy Editor: Chuck Hutchinson

Technical Editors: Shobhit, Flavio Correa

Editorial Assistant: Cindy Teeters

Designer: Chuti Prasertsith

Composition: codeMantra

Indexer: Timothy Wright

Proofreader: Barbara Mack

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Pearson's Commitment to Diversity, Equity, and Inclusion

Pearson is dedicated to creating bias-free content that reflects the diversity of all learners. We embrace the many dimensions of diversity, including but not limited to race, ethnicity, gender, socioeconomic status, ability, age, sexual orientation, and religious or political beliefs.

Education is a powerful force for equity and change in our world. It has the potential to deliver opportunities that improve lives and enable economic mobility. As we work with authors to create content for every product and service, we acknowledge our responsibility to demonstrate inclusivity and incorporate diverse scholarship so that everyone can achieve their potential through learning. As the world's leading learning company, we have a duty to help drive change and live up to our purpose to help more people create a better life for themselves and to create a better world.

Our ambition is to purposefully contribute to a world where

- Everyone has an equitable and lifelong opportunity to succeed through learning
- Our educational products and services are inclusive and represent the rich diversity of learners
- Our educational content accurately reflects the histories and experiences of the learners we serve
- Our educational content prompts deeper discussions with learners and motivates them to expand their own learning (and worldview)

While we work hard to present unbiased content, we want to hear from you about any concerns or needs with this Pearson product so that we can investigate and address them.

Please contact us with concerns about any potential bias at https://www.pearson.com/report-bias.html.

Figure	Credit/Attribution
Figure 3-12, Figure 12-22, Figure 12-34, Figure 13-16, Figure 13-19, Figure 13-20, Figure A-1, Figure A-3, Figure A-5, Figure A-11, Figure A-12, Figure A-16 Figure A-25, Figure A-28	Microsoft Corporation
Figure 6-2-Figure 6-4, Figure 9-5, Figure 6-11, Figure 6-12	Wireshark
Figure 7-4	nostal6ie/Shutterstock
Figure 7-5	Terry Leung/Pearson Education Asia Limited
Figure 7-6	Monkey Business Images/Shutterstock
Figure 11-2	Bluepixel Technologies
Figure 12-11-Figure 12-13, Figure 12-20, Figure 12-21, Figure 12-24, Figure 13-7, Figure 13-8	Internet Society
Figure 12-23, Figure 12-26, Figure 12-27, Figure 12-30, Figure 12-31, Figure 12-32	Postman, Inc
Figure 13-3, Figure 13-10, Figure A-17-Figure A-20, Figure A-29, Figure A-30	GitHub, Inc
Figure 13-26-Figure 13-30	Grafana Labs
Figure A-6-Figure A-10, Figure A-13-Figure A-15, Figure A-23, Figure A-24, Figure A-26, Figure A-27, Figure A-31, Figure A-32, Figure A-37	Apple, Inc

About the Authors

Simone Arena is a principal technical marketing engineer (TME) within the Cisco Enterprise Networking & Cloud group and is primarily focused on enterprise network architecture and on all things related to wireless and mobility. Simone is based in Italy and is a Cisco veteran, having joined Cisco in 1999. Throughout the years, Simone has covered multiple roles at Cisco, starting as a software engineer working with Catalyst switching platforms, to consulting system engineer in the field, to TME within different teams (Enterprise Solution Engineering, Wireless Business Unit, Enterprise Networking and Cloud, and now Networking Experiences Group). Today Simone is the lead TME architect for Catalyst Wireless, and his time is split between helping customers and partners design the best solution that fits their needs and engineering and product management, trying to evolve and improve the products and solutions. Simone is a Distinguished Speaker at Cisco Live and has spoken at Cisco Live events all over the world for several years. Besides wireless and networking, Simone has two passions: his family, with his two daughters Viola and Anita; and Fiorentina, the best soccer team in the world…no question. In his spare time, Simone enjoys listening to music, especially through his new tube amplifier (simply awesome!).

Francisco Sedano Crippa, CCIE No. 14859, joined Cisco in 2006. After some years at TAC supporting voice solutions and as a system engineer working with service providers, he moved to the development side, where he worked on routing, datacenter and, during the past 10 years, as a technical leader on the Wireless Controller development team, focused in serviceability, location services, programmability, and cloud. He's a Cisco Live speaker and is passionate about DevOps and automation, and he is now working on architecting next-generation cloud-based lab services. When not working, he spends his time building a full-size Boeing 737 simulator in his basement and enjoying his other passion: his daughter, Scarlett, and son, Marco, and his wife, Isabel.

Nicolas Darchis, CCIE Wireless No. 25344, joined the Wireless and AAA Cisco TAC team in Belgium in 2007, where his main focus was troubleshooting wireless networks, wireless management tools, and security products. Since 2016, Nicolas has been working as a technical leader for wireless at the same technical assistance center in Brussels; he has shifted a big part of his focus to improving product serviceability of new and upcoming products, as well as new software releases. He is also a major contributor to online documentation of Cisco wireless products and has participated in many of the wireless "Ask the Expert" sessions run by the Cisco support community. Nicolas has been a CCIE Wireless No. 25344 since 2009 and, more recently, he has achieved CWNE No. 208.

Sudha Katgeri, CCIE No. 45857, is a technical leader in services for Enterprise Wireless and has been with Cisco since 2006. Besides supporting customer escalations, Sudha collaborates with Customer Experience (CX), Enterprise Networking (ENB) Escalation, Engineering, and Product Management to improve product quality and serviceability in the next-generation Catalyst wireless stack. Sudha has a CCIE in Wireless (#45857) and is an author and contributor to Wireless TAC Innovation Tools like Wireless Config Converter, CLI Analyzer, and several documents on cisco.com.

About the Technical Reviewers

Shobhit is a principal engineer in Enterprise Wireless Engineering at Cisco Systems. He has over 15 years of experience working on Enterprise Wireless LANs and Mobility, across a range of products, and has been involved with the Catalyst 9800 wireless LAN controllers since they were conceptualized. He has extensively worked on the architecture, design, and implementation of the software, which runs on Catalyst 9800 WLCs, and he continues to be passionately involved with Catalyst 9800 adoption.

Shobhit lives in Bangalore, India, with his wife, Shweta, and their eight-year-old daughter, Snehi.

Flavio Correa, CCIE Wireless No. 38913, joined Cisco in 2008 and is based in Brazil. He is a lead technical solutions architect for Latin America with more than 20 years of experience designing and deploying wireless technologies. He holds a bachelor's degree in electrical and electronics engineering from Mackenzie University and an MBA in data science from the University of São Paulo.

Dedications

I would like to dedicate this book to my father; I know he is watching me from up there, and he would be very proud. I also want to thank my family (all of them, all members of my big Italian family!) for pushing me to be a better man every day and cheering for me when I need it. Final mention to my two dogs, Barney and Mia, and the two cats, Harry and Ginny, that keep me company in my long hours in the office.

—Simone Arena

I would like to first dedicate this book to my parents. When I started my studies, I often went with them to buy books to prepare for CCIE, and we dreamed about writing one someday. It has been possible, thanks to all their support. I'm also lucky to have the best friends ever, especially Abel, Oscar, and Jaci. We shared so many long study nights and some unforgettable memories. And my wife, Isabel, who has supported me in every (usually crazy) idea I have. And, my son, Marco, and daughter, Scarlett. Without them, the book would have probably been published earlier, but the world would be less awesome.

—Francisco Sedano Crippa

I would like to dedicate this book to my family, who have supported me on every step of this long journey: Caroline, my wife, and Maxime, Emeline, and Capucine, my children who accepted I spent time writing instead of playing with them. I would like to particularly thank all the people in Cisco Engineering who love the product they work on and are always happy to provide answers, help, and confirmation about certain behaviors and are always eager to hear feedback about how we could improve the product. Giving names would mean I probably would forget some, so I prefer to thank them privately myself and collectively here.

—Nicolas Darchis

I dedicate this book to my family, my parents for inspiring me to follow my dreams, and my husband and kids for their unwavering love and support, without which I could not have fulfilled my goals or been part of this book. I also want to thank my coauthors for their understanding and encouragement throughout this undertaking.

—Sudha Katgeri

Acknowledgments

Special thanks to our awesome technical reviewers: Shobhit and Flavio Correa. They both helped us tremendously throughout this book journey not only with their corrections and attention to details but also with their suggestions on the content, identifying missing use cases and hence improving the quality of the book.

We would like to thank our management leadership in Cisco for supporting us during the lengthy process of writing a book. Thank you to the EMEA and US CX leadership: Matthew Batson, Kathy Ferguson, Wes Moss; and EN Product Management leadership: Greg Dorai, Chandan Mehndiratta, and Muhammad Imman.

This book couldn't have been possible without the support of many people on the Cisco Press team. Thanks to Nancy Davis, executive editor; her enthusiasm and dedication were instrumental in getting the book done. Eleanor Bru, development editor, did an amazing job in the technical review cycle, and it has been an absolute pleasure working with you. Also, many thanks to the numerous unknown soldiers at Cisco Press working behind the scenes to make this book happen.

Contents at a Glance

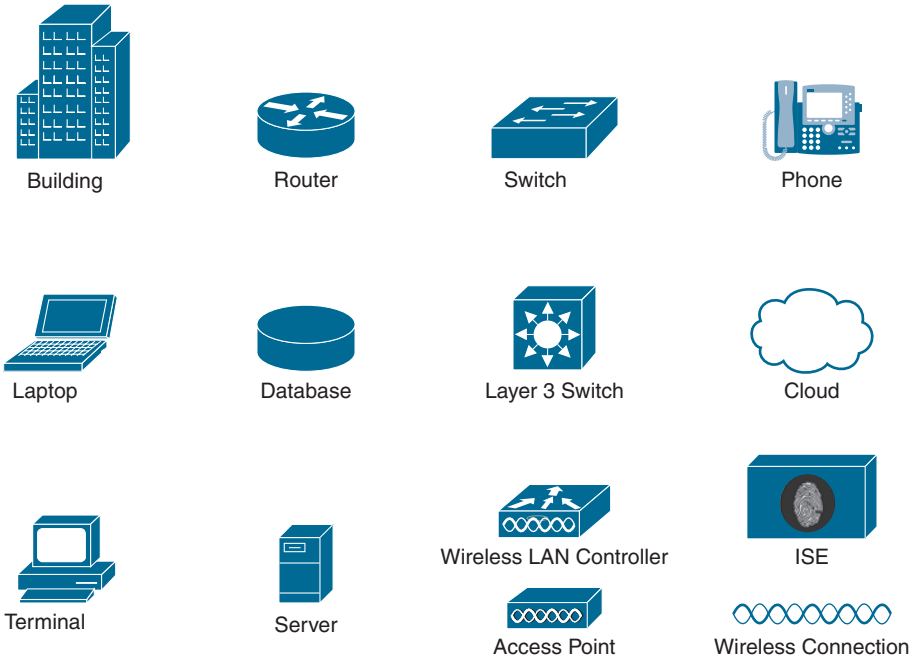
Contents

Reader Services

Register your copy at www.ciscopress.com/title/ISBN for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to www.ciscopress.com/register and log in or create an account*. Enter the product ISBN 9780137492329 and click Submit. When the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box that you would like to hear from us to receive exclusive discounts on future editions of this product.

Icons Used in This Book



Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in Cisco's Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italics* indicate arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets [ ] indicate optional elements.
- Braces { } indicate a required choice.
- Braces within brackets [{ }] indicate a required choice within an optional element.

Introduction

Wireless networks are continuously evolving: the technology is changing at a very fast pace from Wi-Fi 4/5 to Wi-Fi 6/6E and to Wi-Fi 7, which is not too far away on the horizon. The applications and the services enabled on the Wi-Fi network are evolving as well, becoming more and more business critical with stringent requirements on performance and latency. The number of devices that needs to be supported is increasing, and high-density deployments are becoming the norm.

To meet these fast-changing requirements, you need a wireless network that can quickly adapt, that is secure and can perform and scale to higher standards, and that can be easily and programmatically managed. This is the reason behind the introduction of the next-generation wireless controller, the Catalyst 9800 (C9800), which is a critical element of the new Catalyst wireless stack.

The C9800 is based on a new modern and secure operating system, Cisco IOS-XE, and it's built from the ground up for Cisco intent-based networks to deliver on the next wave of wireless innovations.

Goals and Methods

The goal of this book is to educate wireless professionals on how to design, deploy, and manage a Catalyst wireless network built using the new Catalyst 9800 wireless LAN controller, providing practical tips and recommendations.

Who Should Read This Book?

This book is for all readers who are passionate about wireless and want to learn how to design, deploy, and manage a Cisco Catalyst wireless network. The book has been written with three groups of readers in mind:

Technical Decision Makers (TDM): The book familiarizes TDMs with a deep technical view of the Catalyst 9800 wireless controller and helps you in addressing your specific business needs.

Network architects: The book provides technical details on the software and hardware architecture of the Catalyst wireless solution so that you can better understand how to design your own network and implement the features and functionalities you need.

Network operators and IT professionals: The book provides useful tools and tips for setting up, configuring, and monitoring and troubleshooting the network, making your work as a network operator easier.

How This Book Is Organized

The book layout follows the lifecycle of the design and deployment of a Catalyst wireless network using a C9800 wireless controller and hence is divided into three parts:

1. Designing and bringing up the network: Day Zero
2. Configuring and deploying: Day One
3. Monitoring and troubleshooting: Day Two/Day N

Most of the theory is laid out in advance so you can understand the concepts before you see the practical aspects of each topic. We recommend you go through the chapters sequentially to get the full benefit of the book.

Book Structure

In part one, you find the following chapters:

Chapter 1, Cisco C9800 Series: This chapter introduces the Catalyst 9800 wireless controller, its benefits, and its main characteristics. The chapter clarifies the reasons for a new wireless controller for intent-based networking and the role it plays in a cloud-based management network. The chapter also describes the flexible options to manage the C9800, the licensing model, and how this new wireless controller fits into the next-generation wireless stack.

Chapter 2, Hardware and Software Architecture of the C9800: This chapter covers the software architecture of the Catalyst 9800, the reasons behind using a unified IOS-XE for all Cisco enterprise platforms, and the benefits of a scalable architecture.

Part two has the following chapters:

Chapter 3, C9800 Configuration Model: This chapter introduces the new configuration model available for the Catalyst 9800 wireless controller. Important design considerations and best practices are illustrated to get the best performance and stability out of your C9800-based wireless network.

Chapter 4, C9800 Deployment and Installation: This chapter illustrates the different deployment modes supported for the private and public cloud and describes how to bring up your C9800 to an operational state. It then suggests different methods to easily configure your first WLAN on a Catalyst wireless network.

Chapter 5, Security: This chapter covers all the security aspects of the Catalyst 9800 controller, focusing on AAA operations, the use of access control lists (ACLs) to restrict client traffic or to protect the controller management plane, and rogue detection and the Wireless Intrusion Prevention System (WIPS).

Chapter 6, Mobility and Client Roaming: This chapter explains in detail the concept of seamless client roaming in an enterprise deployment. It also describes optimizations such as Fast Secure Roaming, 802.11k, 802.11v, and 802.11r-FT. Deploying mobility and related

roaming optimizations on the C9800, and in co-existence with existing AireOS WLCs, is key to enabling a successful adoption of the C9800.

Chapter 7, RF Deployment and Guidelines: This chapter covers basic antenna concepts and radio resource management (RRM) features and functionalities and provides the tuning and recommendation details you need to have your wireless network running like clockwork.

Chapter 8, Multicast and Multicast Domain Name System (mDNS): This chapter describes the configuration and optimizations available on the Catalyst 9800 wireless controller to deliver broadcast, multicast, and mDNS traffic effectively and efficiently.

Chapter 9, Quality of Service (QoS): The C9800 is based on IOS-XE and leverages the Cisco Modular QoS CLI (MQC) to define and implement an end-to-end architecture for QoS. The chapter describes the design and deployment of the C9800 "trust DSCP" QoS model and provides best practices.

Chapter 10, C9800 High Availability: This chapter describes high availability capabilities of the Catalyst 9800 wireless controller. The C9800 provides stateful switchover (SSO) to allow subsecond failover for access points (APs) and clients. The chapter delves into the design and deployment of the network to leverage redundancy features for the C9800.

Part three features the following chapters:

Chapter 11, Cisco DNA Spaces Integration and IoT: This chapter covers at a high level the value-added services that the wireless network can provide, mainly via Cisco DNA Spaces. The chapter explains the integration of DNAS with the Catalyst 9800 and APs through the Cisco DNA Spaces connector, the protocols involved, and the solutions offered.

Chapter 12, Network Programmability: This chapter describes network programmability concepts and the protocols used in modern network programmability, such as NETCONF and RESTCONF. It describes the YANG data models used in the Cisco 9800. It also presents examples illustrating how Python can be used to query the Cisco 9800 operational model and how to create site tags using RESTCONF.

Chapter 13, Model-Driven Telemetry: This chapter describes how you can benefit from model-driven telemetry on the Catalyst 9800 wireless controller and how to integrate the rich data models available with open-source tools.

Chapter 14, Cisco DNA Center/Assurance Integration: This chapter gives an overview of how Cisco DNA Center Assurance works and how it can help you in troubleshooting your Catalyst wireless networks. The chapter focuses on the Catalyst 9800 integration with Cisco DNA Center.

Chapter 15, Backing Up, Restoring, and Upgrading Your C9800: This chapter covers different operational aspects of the C9800, including how to save the configuration on the controller, how to back it up to an external location, and how to restore the configurations in case of problems on the device or in case of complete device replacement.

Chapter 16, Troubleshooting: This chapter explains the various monitoring, debugging, tracing, and packet-capturing features that are native to the C9800 and introduces tools that are available on-the-box and offline that help with data collection and analysis to determine the root cause of problem scenarios.

Appendix A, Setting Up a Development Environment: In this extra hands-on appendix, you learn how to install and use the tools needed to work with C9800 programmability. This appendix offers an overview of a modern development environment, including Git and container-based environments.

Chapter 1

Cisco C9800 Series

The Cisco Catalyst 9800 series is the next generation of wireless controllers for the enterprise. Based on the Cisco IOS-XE operating system, the Catalyst 9800 is built from the ground up for intent-based networking to deliver on the next wave of wireless innovations and to address new requirements coming from emerging standards like Wi-Fi 6, Wi-Fi 6E, and Wi-Fi 7 in the near future.

Cisco Catalyst 9800 series wireless controllers integrate over 20 years' worth of Cisco radio frequency (RF) excellence with a modern, scalable, and programmable operating system to create a best-in-class wireless network. Together with Catalyst Access Point, Cisco DNA Center, and Cisco DNA Spaces, it provides the next generation of wireless experience and addresses the enterprise's evolving and growing digitization needs.

Cisco Catalyst 9800 series wireless controllers are feature-rich and enterprise-ready to power your business-critical operations and transform end-customer experiences. The main advantages are as follows:

- **High availability:** In-Service Software Upgrade (ISSU), seamless software upgrades and updates enabled by hot and cold patching, and stateful switchover (SSO) keep your clients and services always on during both planned and unplanned events.
- **Security:** You are able to secure the air, devices, and users with the Cisco Catalyst 9800 series. The wireless infrastructure becomes the strongest first line of defense with Cisco Encrypted Traffic Analytics and software-defined access (SDA). The controllers come with built-in security: secure boot, runtime defenses, image signing, integrity verification, and hardware authenticity.
- **Flexibility:** You can deploy anywhere to enable wireless connectivity everywhere. Whether on-premises, in a public or private cloud, or embedded on a switch or access point, the Cisco Catalyst 9800 has multiple deployment and scale options to best meet your organization's needs.
- **Open and programmable framework:** Built on a modular operating system, Catalyst 9800 controllers feature open and programmable APIs that enable automation of

your Day Zero (Day 0) to Day N network operations. Model-driven streaming telemetry provides deep insights into the health of your network and clients.

Cisco Catalyst 9800 series wireless controllers are available in multiple form factors:

- **Cisco Catalyst 9800-L**: This compact wireless controller appliance is perfect for small- to medium-sized network deployment. Data ports can operate in 1 GE and 10 GE mode, supporting different SFP/SFP+ transceivers and up to 5 Gbps of throughput (up to 10 Gbps with the optional performance license). It scales to 500 access points and 10,000 clients with a performance license.
- **Cisco Catalyst 9800-40:** This one rack unit fixed wireless controller appliance scales from medium to large deployments. Data ports can operate in 1 GE and 10 GE mode, supporting different SFP/SFP+ transceivers and up to 40 Gbps of throughput. It scales to 2000 access points and 32,000 clients.
- **Cisco Catalyst 9800-80**: Wireless controller appliance modular uplinks provide flexible connectivity options supporting 10 GE, 40 GE, and 100 GE QSFP hot-swappable. Fixed data ports can operate in 1 GE and 10 GE mode, supporting different SFP/SFP+ transceivers and up to 80 Gbps of throughput. It scales to 6000 access points and 64,000 clients.
- **Cisco Catalyst 9800-CL for Private Cloud**: This cloud controller with deployment flexibility offers the hypervisor of your choice (KVM, VMware ESXi, Microsoft Hyper-V, or Cisco ENCS). It scales from 1000 to 6000 access points and up to 64,000 clients.
- **Cisco Catalyst 9800-CL for Public Cloud**: This cloud wireless controller is used for the public cloud of your choice (Amazon Web Services, Google, or Azure Cloud) and your Infrastructure-as-a-Service (IaaS) deployments. It scales from 1000 to 6000 access points and up to 64,000 clients. It is available for FlexConnect deployments only.
- **Cisco Catalyst 9800 Embedded Wireless Controllers (EWC) on Catalyst Access Points**: This embedded wireless controller for the Cisco Catalyst 9100 access point can be positioned for small standalone deployments. It scales to 100 access points and 2000 clients. It is available for FlexConnect deployments only.
- **Cisco Catalyst 9800 Embedded Wireless Controllers (EWC) on Catalyst Switches**: This embedded wireless controller is used with the Cisco Catalyst 9000 series switches for software-defined access deployments. It scales up to 200 access points.

Why Cisco C9800?

You might be wondering why Cisco decided to invest in a new generation of wireless local-area network (LAN) controllers, the Catalyst 9800. The AireOS-based controllers have provided an industry-first enterprise class solution for the last 15 years, innovating and driving customers through many iterations of the Wi-Fi standards: from the early days of 802.11n, to 802.11ac, waves 1 and 2, to the most recent 802.11ax standard. So why C9800 and, most importantly, why now? To answer this fundamental question, we

need to introduce intent-based networking as the Cisco strategy for the enterprise and understand how Catalyst 9800 fits into it.

Intent-Based Networking (IBN)

Enterprise customers need to deal with a continuously evolving technology landscape. The pace of change is fast and driven primarily by global trends like mobility, Internet of Things (IoT), and cloud adoption. As a result, networks are becoming more complex, need to change frequently, and must respond to new and sophisticated cybersecurity threats.

Customers need a simpler way to design, deploy, and operate their network to be agile in this evolving digital economy. Cisco's answer to these trends is intent-based networking (IBN).

IBN builds on software-defined networking (SDN) by using a network controller that acts as a central control point for network activity. Such controllers are crucial for network abstraction and automation that lets IT treat the network as an integrated whole. Controller-led networks in all domains (including access, WAN, data center, and cloud) collaborate and extend their benefits throughout the enterprise and help make digital transformation a reality.

The controller relies on an intelligent network that can be programmed through open and secure interfaces so that you can express "intent" (what you want the network to do), and this gets translated and instantiated automatically into a network device optimized for configuration. This is IBN automation. On the other side, the intelligent network needs to continuously capture relevant analytics and efficiently stream it, leveraging streaming telemetry, back to the controller to be analyzed, interpreted, and correlated. This is IBN assurance, which allows you to realize a closed-loop system and verify whether the actual intent has been met. The controller for the enterprise is Cisco DNA Center.

As shown in Figure 1-1, the closed-loop system of Cisco DNA Center leverages two important building blocks: the IOS-XE network operating system and Cisco programmable hardware.

Figure 1-1 *The principles of intent-based networking*

These two pillars of the enterprise architecture allow Cisco to continuously deliver innovation at a solution/application level and bring value to customers. And it's on these two foundation elements that the next generation of Catalyst 9800 wireless LAN controller (WLC) has been built.

Flexible Software

One of the pillars of IBN is Cisco IOS-XE, a modern and secure network operating system embedded in all Catalyst products: wireless, routing, and switching devices. This gives Cisco a great advantage: a network that is not made by silo products but instead built as one network, with a common operating system, a common policy structure, a common automation and assurance framework, with common programmable interfaces. This operating system hugely simplifies how customers consume networking services.

Figure 1-2 lists some of the key characteristic of a modern networking operating system.

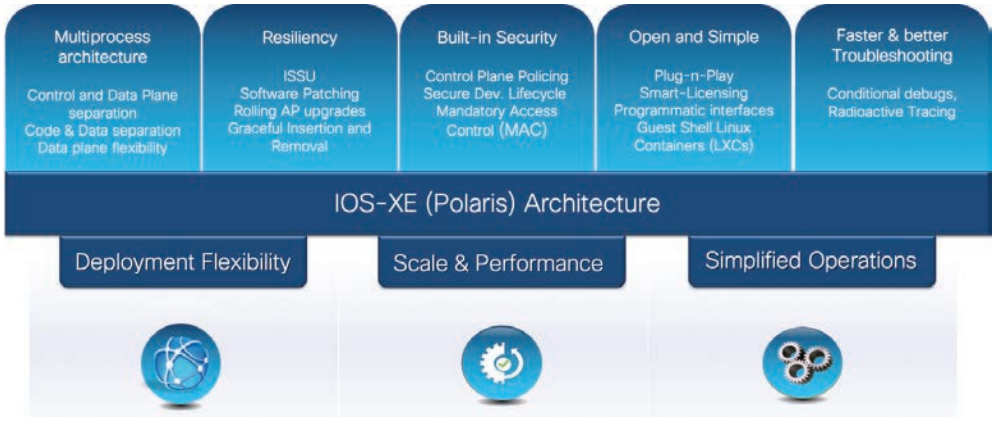


Figure 1-2 *IOS-XE: A modern networking operating system*

Let's look closely at the advantages that Cisco IOS-XE brings to Catalyst 9800:

- **Multiprocess architecture:** Every important function in C9800 is a single process, single thread with separated memory and fault domain: the AP and client Session Manager (known as the Wireless Network Controller Daemon, or WNCd), radio resource management (RRM), Mobility Manager, Rogue Management, and so on. This software architecture not only allows for management, control, and data plane separation and segregation, but also provides a framework for more flexibility, resiliency, and scalability.
- **Resiliency:** A multiprocess architecture and data externalization (both configuration and operational data) allow IOS-XE to provide a much more resilient software architecture. This is the foundation for process restart, process patching, and for In-Service Software Upgrades (ISSUs) and rolling AP upgrades, an RF-based intel-

ligent AP software upgrade mechanism. All of these are important innovations that deliver unprecedented resiliency to your wireless network.

- **Built-in security:** C9800 can also leverage the key security features embedded in IOS-XE, such as control plane policing, secure development lifecycle, and mandatory access control (MAC).
- **Open, simple, and programmable framework:** Features include Plug and Play, zero-touch provisioning (ZTP), Embedded Event Manager, Guest Shell, smart licensing, programmatic interfaces with NETCONF/RESTCONF and Yang models. These are some of the features directly inherited from IOS-XE and now available to Catalyst 9800 customers.
- **Faster and easier troubleshooting:** Conditional debugs and radioactive tracing are two of the main innovations for troubleshooting and serviceability that provide the next level of network insights and deep visibility into users, applications, and device behaviors. When an issue happens and you need to identify the root cause, this information is available "on box," but it's also sent via streaming telemetry to Cisco DNA Center Assurance, which provides a single pane of glass for monitoring your entire network.

As shown in Figure 1-2, all these IOS-XE innovations translate directly into tangible customer advantages: from Day Zero deployment flexibility and ease of use; to Day One scale, performance, and resiliency; to Day Two operation efficiency. Across the entire lifecycle, this network is easy to design, deploy, and operate.

Flexible Hardware

> *People who are really serious about software should make their own hardware.*
> —Alan Kay

This famous quote is from Alan Kay, a computer scientist and pioneer in work on object-oriented programming and graphical user interface (GUI) design. In other words, the intersection of flexible hardware and flexible software really brings value to an organization, especially in the current time of rapid technology and organizational change. Flexible hardware allows organizations to adapt their network over time and adopt new functions, new protocols, and new solutions without compromising on performance (thanks to the hardware acceleration) and without replacing the network hardware. This provides investment protection and maximizes the return on investment (ROI) in their infrastructure.

As illustrated in Figure 1-3, the Catalyst 9800 is built on a programmable application-specific integrated circuit (ASIC), as with the Cisco Quantum Flow Processor (QFP) in the C8000-80 and 9800-40 or the Unified Access Data Plane (UADP) chip in the C9800 embedded in the Catalyst 9000 switches. Flexible silicon in the Catalyst 9800 allows customers to take advantage of the latest networking innovations and at the same time provides industry highest performances in terms of throughput and scale.

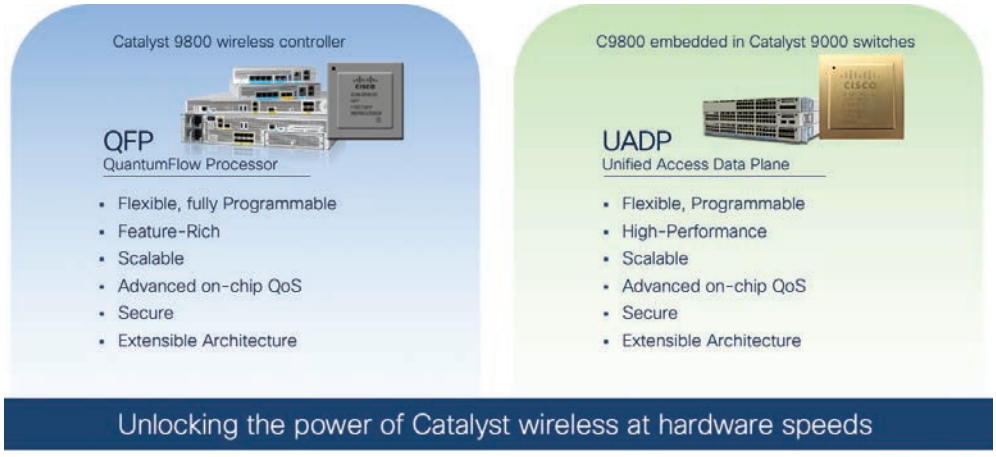


Figure 1-3 *Catalyst 9800 is built on programmable silicon*

With Cisco Catalyst Wireless, the advantages of programmable ASIC are not limited to C9800 but are also extended to the new Wi-Fi 6 Catalyst AP, as shown in Figure 1-4.

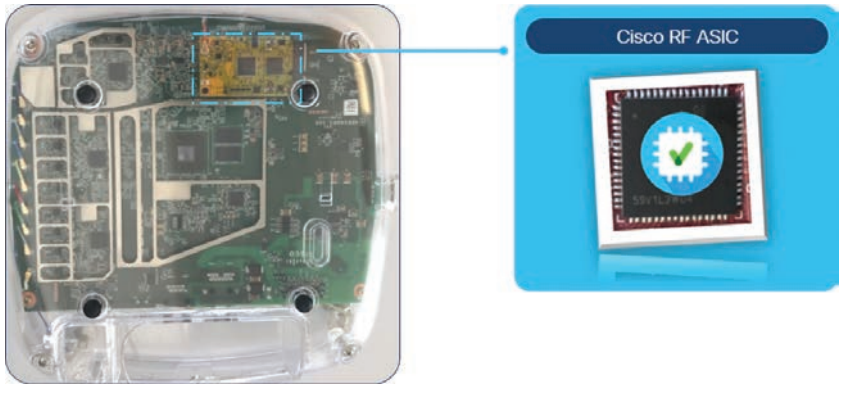


Figure 1-4 *Cisco RF ASIC: Software-defined radio using a Mini-PCIe slot*

The Cisco RF ASIC is a software-defined "radio on a module" whose main purpose is to analyze a large range of frequencies and convert the RF baseband to data, which is then analyzed by the CleanAir engine. Capable of extremely high resolution of 78.125 kHz (at least four times better than the nearest competitor), the module is embedded in the Catalyst 9130, 9120, and 9124 access points. It provides Cisco's RF excellence-related features:

- **CleanAir:** Monitors the spectrum for and identifies non–Wi-Fi sources of interference
- **Zero Wait Dynamic Frequency Selection (DFS):** Allows a channel availability check of the DFS channel; allows immediate use without 60s penalty

- **Dual Filter Dynamic Frequency Selection (DFS):** Provides dedicated radar detection to augment the radio vendor detection algorithms; reduces false detection by 99.9 percent
- **FastLocate:** Provides consistent and fast location updates, without requiring dedicated monitor hardware to capture data traffic
- **Off-Channel RRM:** Provides zero client impact off-channel monitoring for RF management, leaving the client serving on radio 100% of the time on-channel availability

As illustrated in Figure 1-5, the 15 years of wireless innovations and RF excellence that customers have enjoyed from AireOS-based wireless controllers are not lost; everything that has been learned has been brought over to the Catalyst 9800, augmented with the advantages you get from a modern, scalable operating system like IOS-XE and combined with Cisco flexible hardware.

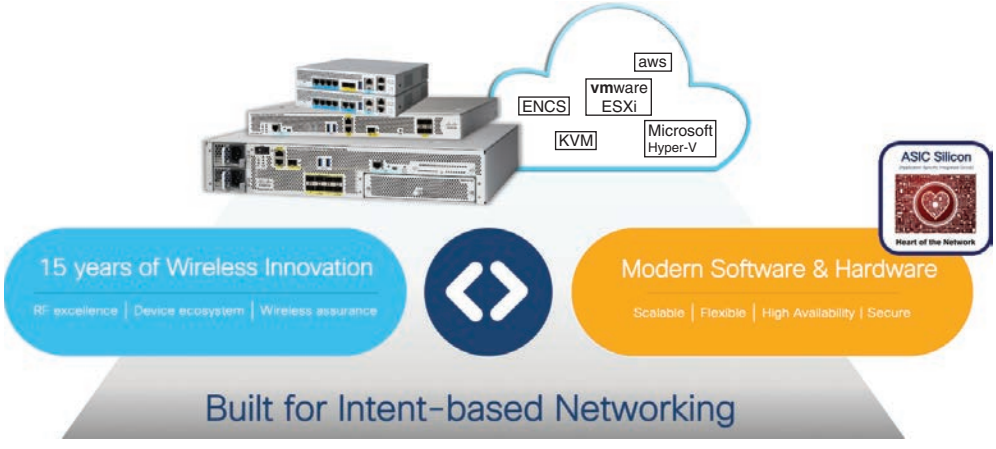


Figure 1-5 *Catalyst 9800 built for intent-based networking*

The Catalyst 9800 has been built from the ground up according to the principles of intent-based networking to deliver on the next wave of wireless standards like Wi-Fi 6, Wi-Fi 6E, and Wi-Fi 7.

The Role of the Wireless Controller in a Cloud Era

The adoption of the cloud is one of the most relevant trends in IT, and networking is no different. Cloud networking is defined as a technology that allows customers to build their networks leveraging cloud-based services. These software and infrastructure services include but are not limited to network management, policy management, and analytics and assurance functions.

The *cloud* here refers to both the public and private cloud. With the public cloud, customers don't own the infrastructure (computing, storage, networking, databases, and so

on), and resources are consumed "as a service", usually through a graphical interface or application programming interface (API) provided by the cloud provider. With the private cloud, customers have exclusive access to dedicated virtualized or physical resources. The resources are either located at on-premises data centers or hosted by a colocation provider.

Here are the main reasons for customers to adopt the cloud and an "as a service" model:

- **Scale:** You have the ability to scale up and down on demand when needed without planning for it up front. This flexibility usually helps to optimize resources.
- **Agility:** This capability usually means bringing new features to market more rapidly because cloud services are usually built independently, often relying on microservices architecture.
- **Up-to-date services:** Services are managed directly by the cloud provider, which will keep them updated as new features are added or issues are fixed.
- **High availability:** Cloud services are redundant in nature because they are deployed across multiple data centers or zones, and all are managed by the provider, so customers don't have to worry about backups and redundancy configuration.
- **Simplicity:** Even if not technically related to the cloud, services in the cloud are usually designed around simplicity because they have to be used by multiple customers. Sometimes this simplicity happens at the cost of customization and control for specific functions that may be desired.
- **Accessibility:** Deploying in the cloud means that the services can be reached from anywhere with an Internet connection.
- **Move to OPEX model:** Some IT organizations may prefer to be less exposed on capital expenditures or expenses (CAPEX) and adopt an operating expenditures or expenses (OPEX)–based model. This is a business reason and not a technical one.

There are no doubts that the cloud can bring a lot of benefits to IT and networking, especially in simplifying operations. Does this mean that everything will move to the cloud and the on-premises infrastructure will be commoditized and not relevant anymore? What is the role of the wireless LAN controller in a cloud-managed network?

To answer these questions, we need to first analyze the functions that a WLC has performed traditionally and see how they should evolve to fit in a cloud management era.

Cisco introduced the AireOS-based WLC in 2005 mainly to address the requirements for scale in rapidly growing wireless networks, and since then, the WLC functions have continuously evolved and become richer. Here is a list of the key functions:

- Access point (AP) image management (no more updating each single AP)
- Radio resource management (RRM; defining and implementing an RF channel and transmitting a power plan for the whole network)
- Configuration management
- Network access server (NAS) for client authentication and authorization

- Data plane termination and anchor point for mobility (to allow roaming at scale and across different subnets, or Layer 3 roaming)
- Consistent client mobility database
- Single point of policy definition
- Single point of ingress to the wired network
- Single serviceability and troubleshooting point
- Communication with external services (AAA, DHCP, NetFlow Collector, and so on)
- Telemetry collection, aggregation, and optimization

These functions can be classified as belonging to the management, control, and data planes and were all initially included in the WLC "box," either a physical or virtual appliance.

From a network architecture perspective, different functions or services have different requirements. Management functions (for example, AP image and configuration management, analytics and assurance) and non–real-time control plane services (for example, RRM, rogue management, policy definition) need to be designed for scale, ease of use, and role-based access. On the other hand, real-time control plane functions (for example, client authentication, roaming management) and data plane services (for example, data plane termination, policy enforcement) should be designed around low latency, data consistency, data optimization, and high density.

As a general wireless architecture principle, it is important to understand where these functions may belong: management and non–real-time control plane functions may be centralized either in the cloud or in a central location where it's easier to deploy at scale. Real-time functions are better deployed on-premises, close to the client devices to fulfill the stringent requirements on latency.

In the design of the next-generation Catalyst 9800, these architecture requirements were taken into consideration, which resulted in a platform that is very flexible and can be deployed according to customer requirements for scale and performance and also according to the type of consumption model—an "as a service" or a "do it yourself" on-premises model.

Here are the main characteristics of the Catalyst 9800 that make it a wireless controller built for the cloud era:

- **Network function virtualization**: With C9800, the wireless LAN controller should not be considered a box anymore, but more a collection of wireless network services that can be deployed in different places on the network: in an appliance (physical or virtual) to have hardware-accelerated, ASIC-based, data plane termination that scales to 80 Gbps of throughput and 6000 access points and 64,000 clients; or embedded in a network device like Catalyst 9000 switches, where the control plane runs on the switch and the data plane is distributed to the wired network. Similarly, for the C9800 embedded in the Catalyst AP for small deployments.
- **API-driven**: Every single configuration for the Catalyst 9800 is available through programmatic interfaces and open configuration models (Yang and OpenConfig models).

- **Model-driven telemetry:** Deep analytics information is captured and streamed efficiently and at scale thanks to streaming telemetry protocols like gRPC/gNMI.
- **Deployable "as a Service":** The Catalyst 9800 is an industry-first wireless controller that can be deployed on a public cloud with an Infrastructure-as-a-Service (IaaS) model. The C9800-CL (where *-CL* stands for cloud) allows you to choose the public cloud of choice—Amazon Web Services (AWS), Google Cloud Platform (GCP), or Microsoft Azure—and enjoy the benefits of the cloud for the compute, storage, and networking resources. At the same time, it allows you to retain full control of your wireless controller services' software releases and configuration as if it were deployed on-premises.

Customer adoption of cloud services varies based on the specific requirements and needs. That's why it's important to provide a wireless architecture that is flexible and can adopt to different deployment scenarios. For example, some customers may require high throughput and fast secure roaming across Layer 3 boundaries, in large roaming domains like the ones found in hospitals and in manufacturing plants or large warehouses. They call for real-time controller plane mobility functions and data plane termination functions to be located on-premises close to the user devices to reduce latency. Other customers may need to reduce the data center footprint and move all possible workloads in the public cloud; this would be the case for a retailer with multiple small branches with few access points that do not actually need the WLC functions to be onsite and hence could benefit from AWS, GCP, or Azure installation. The Catalyst 9800 allows you to design for a flexible architecture that best meets your needs.

Throughout the rest of the book, you learn a lot of Catalyst 9800 functions, what they do, and how to configure and troubleshoot. Before that, however, let's focus on how to manage this platform.

Managing the Cisco C9800

The Catalyst 9800 wireless controller has been built with flexibility in mind, and this is also reflected in the different options that you have to manage the platform, as illustrated in Figure 1-6. Let's analyze each option in more detail.

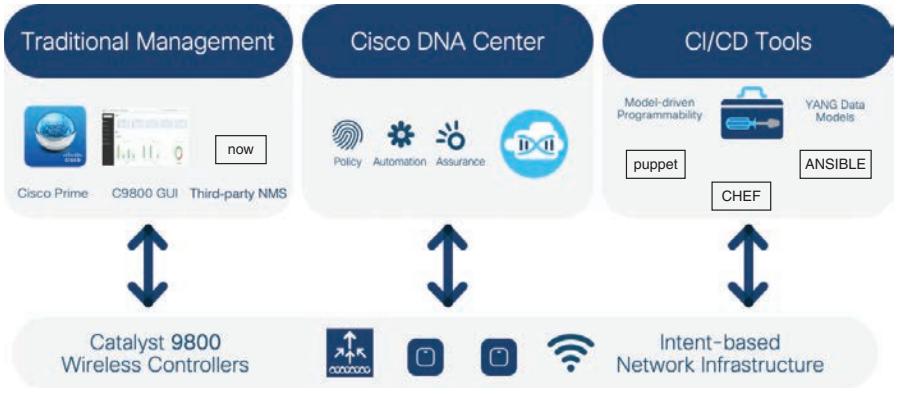


Figure 1-6 *Catalyst 9800 flexible management options*

Traditional Management Tools

You may decide to utilize traditional SNMP-based third-party network management systems (NMSs) to manage the Catalyst 9800. These systems are fully supported, and you can look up the available management information bases (MIBs) in the Cisco Feature Navigator tool and select the desired platform, as illustrated in Figure 1-7.

Figure 1-7 *Cisco Feature Navigator*

This method offers only the base functionalities and doesn't really exploit the full capabilities of the Catalyst 9800, when compared with other Cisco traditional management tools like the integrated graphical user interface and Cisco Prime Infrastructure.

"On Box" Management

There are two ways to manage the Catalyst 9800 directly "on box":

- Text-based command-line interface (CLI)
- Web-based graphical user interface or WebUI

If you know the IOS-XE CLI, now the same interface is available on the wireless controller as well. Some people find the CLI easier to use than WebUI because it allows you to simply copy and paste, to complete commands, to set alias commands, and so on. As with Cisco routers and switches, there are three ways to access the CLI: the local console, remote Telnet, and Secure Shell (SSH).

To log in to the box and configure it remotely, you first need to define a local username with the right privileges. The following CLI shows the minimal recommended configuration to create an administrator user (with privilege 15) and enable SSH:

```
username admin privilege 15 secret 0 password
hostname name
ip domain name domain-name
crypto key generate rsa modulus 2048 label label-name
ip ssh rsa keypair-name label-name
line vty 0 4
   transport input ssh
   login local
```

Starting with release 17.4, the Catalyst 9800 provides a startup wizard via the console CLI to take you through a step-by-step flow and configure the basic settings all at one time. This capability is available only for an "out-of-the-box" scenario where the C9800 has never been configured before or after a factory reset.

If you want a more graphical approach to management, Cisco has also invested in the design of a brand-new graphical user interface for the Catalyst 9800, making it intuitive and easy to use. The new GUI is built on modern front-end web technology to make it light and fast. It features key tools for making a configuration intuitive and for troubleshooting and gathering important insights on your network.

If you're new to Catalyst 9800, the Web interface features a few setup tools to help you get started with the configuration of the most common and important settings you might need to use. You have these different options:

- You can use the WLAN Wizard. Setting an SSID is definitely one of the first tasks you will have to do to get your wireless network up and running. From release 17.5 onward, the C9800 GUI provides a step-by-step flow to configure the most common WLAN profiles and its associated policy for different deployment modes (Local, FlexConnect, and so on). Figure 1-8 shows how you can set up a PSK SSID. Check out the CLI preview, which is very useful.

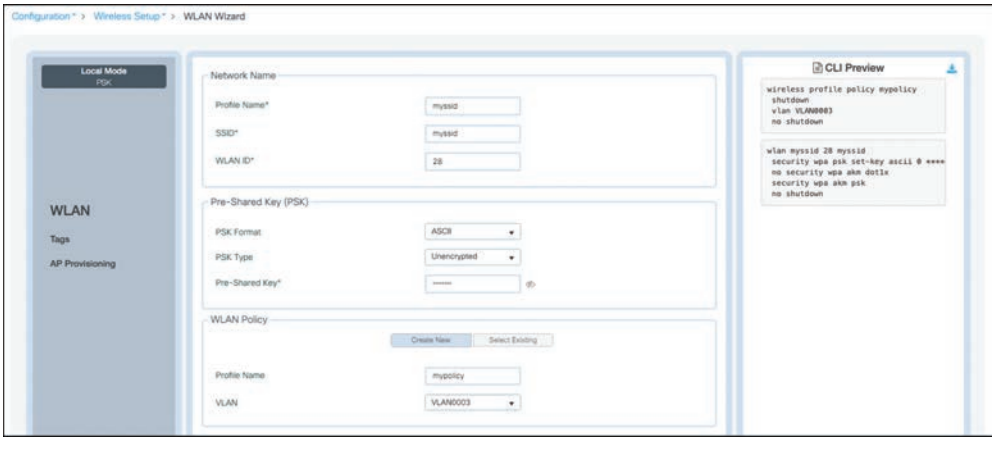


Figure 1-8 *Setting up a PSK SSID with the WLAN Wizard*

- If you need to understand more about the configuration constructs used in C9800, then it's recommended you go through the Advanced Setup Wizard because it introduces you to profiles and tags and takes you through the explicit steps for configuring a wireless network, as illustrated in Figure 1-9. You can find it under **Configuration > Wireless Setup > Advanced**.

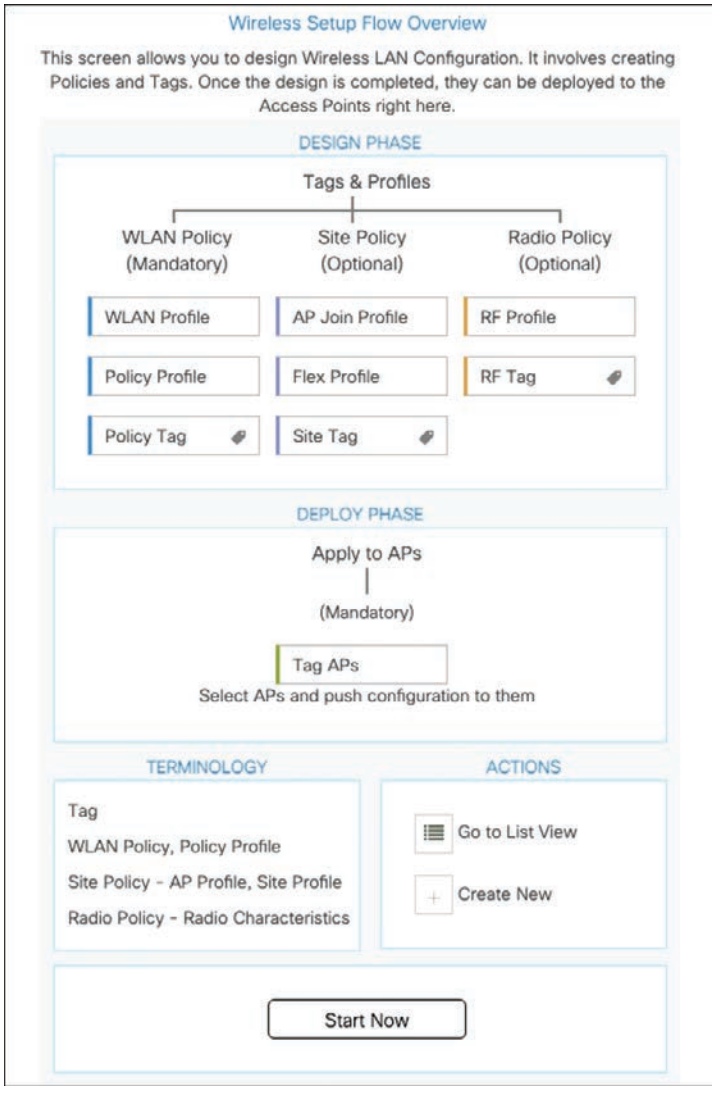


Figure 1-9 *The Advanced Setup Wizard*

- Last but not least, if you have any configuration concerns, you can rely on the Guided Assistance tool. On any page of the Catalyst 9800 GUI, you can click the small banner in the bottom-right corner to launch the tool. At that point, you can type what you're looking for and follow the online instructions, as illustrated in Figure 1-10. This tool also provides an explanation of the different fields you have to configure.

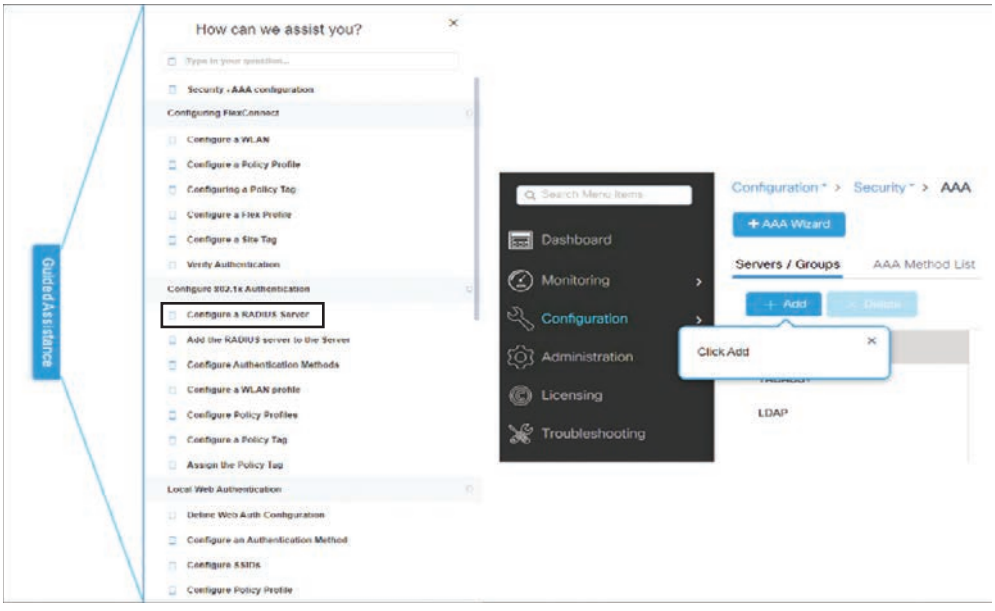


Figure 1-10 *Guided Assistance tool*

If your intent is to dig deeper into an issue that you might have on the network, the Catalyst 9800 WebUI also offers a new and useful troubleshooting tool in the Troubleshooting section of the dashboard. For example, the Packet Capture tool allows you to capture traffic on any interface (wireless management interface, port channel, or physical ports) in an easy and quick way. Just go to the Troubleshooting tab, select **Packet Capture**, and define your capture, as shown in Figure 1-11.

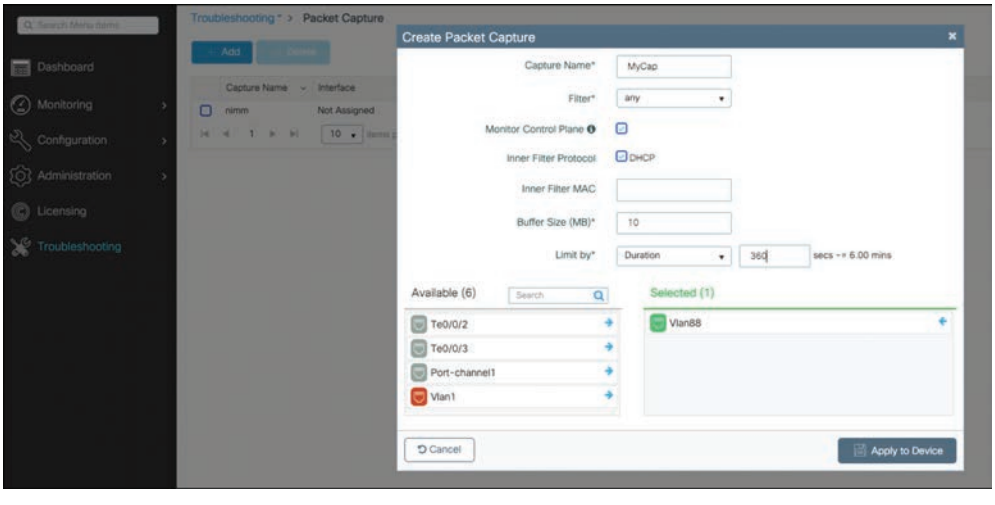


Figure 1-11 *Defining a Packet Capture on C9800's GUI*

Packet Capture on the C9800 allows you to look at the control plane traffic as well, so you can see the messages sent to and from the access point that are DTLS encrypted and hence not visible with a standard Packet Capture on a switch. You can start and stop the capture directly on the GUI and then download the capture file directly on your desktop.

There are some important best practices that you need to keep in mind as you utilize the GUI. The Catalyst 9800's Web GUI leverages Virtual Teletype (VTY) lines for processing HTTP requests. At times, when multiple connections are open, the 15 VTY lines set by the device (the default number) might get exhausted. Therefore, it is strongly recommended that you increase the number of VTY lines to 50. Use the following configuration commands to do this:

```
C9800(config)# line vty 16-50
```

Another useful recommendation is to configure the service tcp-keepalives to monitor the TCP connection to the box. In this case, use the following commands:

```
C9800(config)# service tcp-keepalives-in
C9800(config)# service tcp-keepalives-out
```

Starting with release 17.3, you are able to configure HTTP/HTTPs independently for WebUI access and for portal redirection of client web authentications. For more secure access to the box, it is recommended you disable HTTP for WebUI access. You can go to **Administration > Management > HTTP/HTTPs/NETCONF** and enable it, as shown in Figure 1-12. On the same page, it's also important to explicitly associate a trustpoint to be used for HTTPs connections.

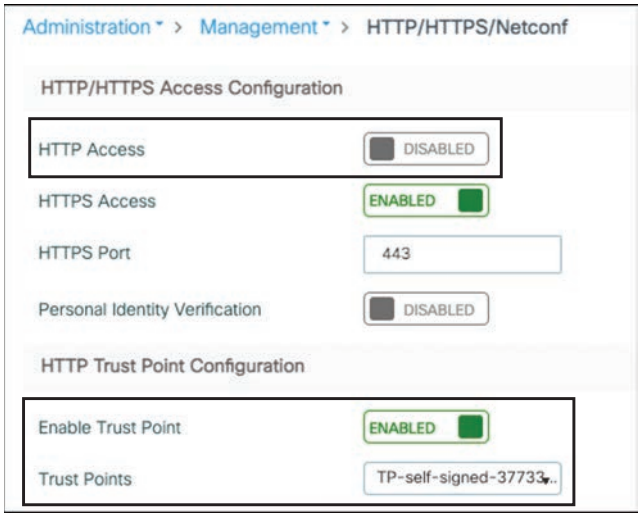


Figure 1-12 *Disabling HTTP web access and selecting a specific trustpoint for HTTPs*

You can find other important recommendations in the Cisco Catalyst 9800 Series Configuration Best Practices online (see the "References" section for details).

Cisco Prime Infrastructure

Cisco Prime Infrastructure (PI) has evolved throughout the years to become a complete management system for enterprise customers, covering the branch to the data center. PI simplifies the management of wireless and wired networks and offers Day 0 and Day 1 provisioning, as well as Day N assurance. Figure 1-13 provides a view of the Prime dashboard.

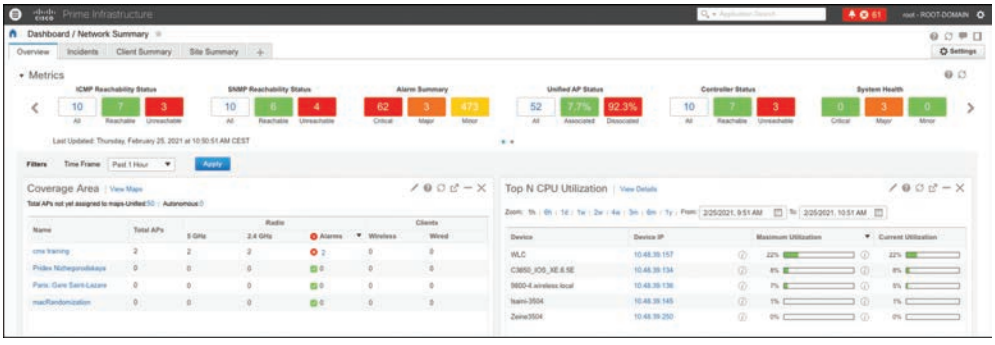


Figure 1-13 *Prime Infrastructure dashboard*

Prime Infrastructure leverages CLI and SNMP information to verify reachability and to add the device to its inventory. It also uses SNMP to push configuration templates as well as support SNMP traps for access points and client events. However, for PI to gather access points and client statistics, NETCONF is leveraged because it's more efficient and allows telemetry to scale better.

For Prime Infrastructure to configure, manage, and monitor the Catalyst 9800, PI needs to be able to access it via SSH, SNMP, and NETCONF. This means that you need to do some preliminary configuration on C9800 itself before PI can manage it. These are the protocols and ports being used in the communication:

- SSH access (recommended over Telnet) is on TCP port 22.
- All configurations are pushed via SNMP templates using UDP port 161 or CLI over SSH/Telnet.
- Operational data for the C9800 box itself is obtained via SNMP UDP port 162.
- AP and client operational data leverages streaming telemetry.

- Prime Infrastructure to C9800 direction: TCP port 830 is used to push telemetry configuration using NETCONF.
- C9800 to Prime Infrastructure direction: TCP port 20830.

Next, you need to follow these steps:

Step 1. Add SNMP configuration: enable SNMP and add community strings for SNMPv2 (a read and write community is required). You can find this on the GUI under **Administration > Management > SNMP**, as shown in Figure 1-14.

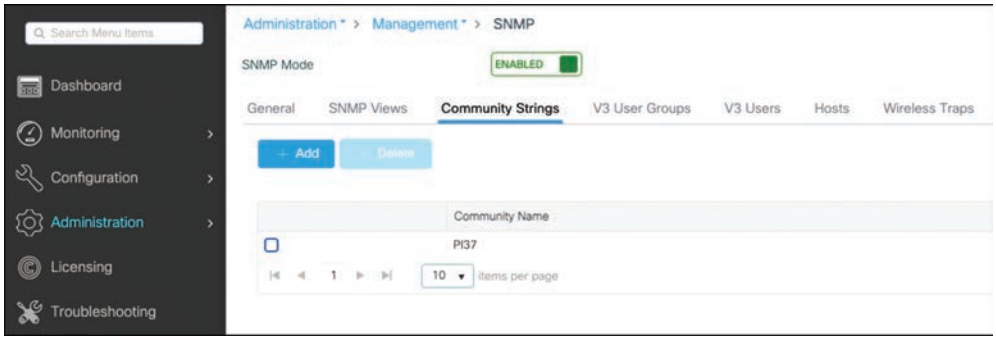


Figure 1-14 *SNMP configuration of the C9800*

You also can configure SNMP via the CLI. Here is an example for SNMPv3 configuration:

```
C9800(config)#snmp-server view view-name iso included
C9800(config)#snmp-server group group-name v3 auth write
view-name
C9800(config)#snmp-server user username group-name v3 auth
{md5 | sha} password priv {3des | aes | des} {optional for
aes 128 | 192| 256} privacy-password
```

Step 2. Enable NETCONF under **Administration > Management > HTTP/HTTPs**, as illustrated in Figure 1-15.

If you already have authentication and authorization settings on the box (aaa new-model is configured), you also need to make sure you specify the repository (local or remote) where the credentials are validated, as shown in the following commands:

```
C9800(config)#aaa authorization exec default local|radius|
tacacs group
C9800(config)#aaa authentication login default local|radius|
tacacs group
```

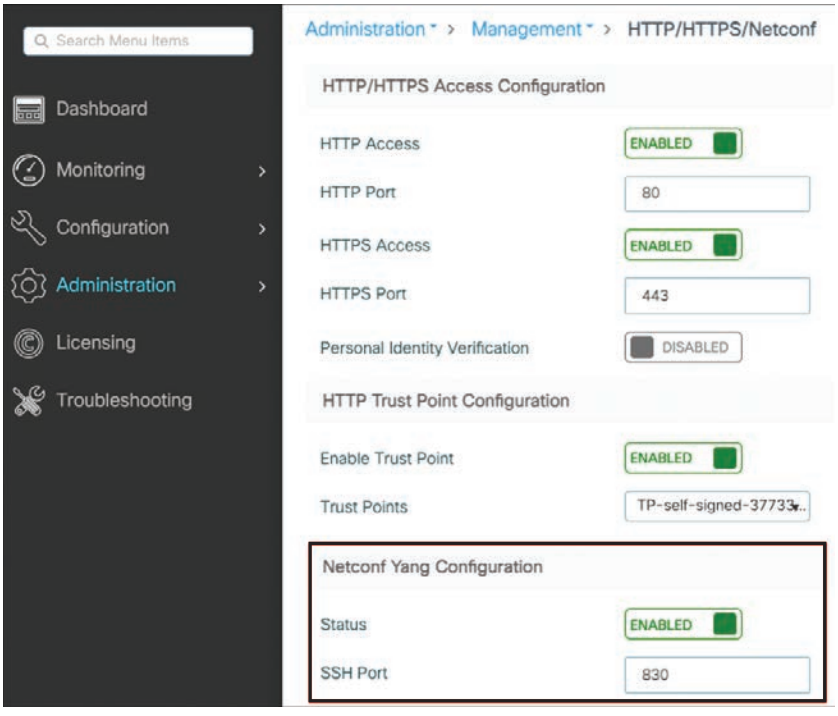


Figure 1-15 *Enabling NETCONF on the Catalyst 9800 GUI*

NETCONF on C9800, by design, leverages the aaa default method (and it cannot be changed) for both authentication and authorization. In case you want to define a different method for SSH connections, you can do so under the "**line vty**" command line. But NETCONF keeps using the default exec authorization method.

Step 3. Grab the Wireless Management Interface (WMI) IP address and the administrator password that you have already configured on the box.

One thing to be careful of when managing the C9800 wireless controller with Prime: when adding to inventory, Prime overwrites the **aaa authentication login default** and **aaa authorization exec default** methods if already configured on the box to point them to the local database. This means if you are using TACACS, you lose CLI access after adding the C9800 to Prime. Keep this in mind and revert those aaa configurations back and make them point to TACACS if that is your preference.

At this point, you are ready to go to Prime Infrastructure, add the C9800 using the WMI interface as the reference IP address, and start managing it. This discussion is beyond the scope of this book, but you can check the configuration guide as a starting point.

Cisco DNA Center

Cisco DNA Center is the enterprise orchestrator and network management platform for the Cisco intent-based network, as illustrated in Figure 1-16.

Figure 1-16 *Cisco DNA Center*

DNA Center represents a new and radically different approach to network management; it provides an intuitive single pane of glass for your entire network and embeds automation, analytics, and assurance into one single platform. DNA Center helps you solve IT challenges by

- **Simplifying operations**: DNA Automation allows you to onboard new network devices in minutes, without onsite support visits. This is key when you have thousands of access points and hundreds of switches to deploy. You can also automate additions, changes, policy provisioning, and software upgrades and make sure your network is compliant.
- **Optimizing the user experience**: DNA Assurance provides a single dashboard to monitor and troubleshoot your networks. It leverages the analytics received from the network and uses advanced artificial intelligence (AI) and machine learning (ML) to proactively monitor, troubleshoot, and optimize your network. For wireless networks, this capability is particularly important because a lot of information can be gathered from wireless clients and access points.

- **Embedding security everywhere:** A unified policy for wired and wireless and network segmentation is easily automated with Cisco software-defined access (SDA). Thanks to the integration with Cisco Identity Services Engine (ISE), Cisco Umbrella, and Cisco Secure Network Analytics (formerly Cisco Stealthwatch), Cisco DNA Center ensures comprehensive security and trusted access everywhere on your network. DNA Policy helps you create policies to scale and control access, route traffic, and prioritize applications, and enforce them consistently across your campus, WAN, and multicloud data center network domains.

A full description of Cisco DNA Center and how to manage the Catalyst 9800 is beyond the scope of this book, but it's important to clarify here what is needed before C9800 can be successfully discovered and managed.

C9800 Prerequisites for Cisco DNA Center

The first step to manage the C9800 with DNA Center is to add it to the inventory. You can do this via the Discovery tool. Here is the list of settings you need to configure on the C9800 before successfully discovering it:

- Configure an admin user with privilege 15.
- Enable SSH on the box.
- Configure a Wireless Management Interface and a route to reach DNA Center.
- Make sure that NETCONF is enabled. The required configuration commands for NETCONF and AAA authorization are

```
c9800(config)#netconf-yang
c9800(config)#aaa new-model
c9800(config)#aaa authorization exec default local
```

  If you're using the AAA server to authenticate the user credentials, make sure that the NETCONF user returned from AAA is defined with privilege 15.
- For discovery, it is recommended you use an IP range with just one IP address (the one for the Wireless Management Interface of the C9800). Using the WMI is preferred and recommended over the service port (SP).
- If you are discovering a C9800 stateful switchover pair, make sure you don't use the Redundancy Manager Interface.

It's also important to keep in mind that DNA Center pushes its own self-signed certificate to the managed devices; the default certificate is **sdn-network-infra-iwan**. When the Catalyst 9800 has more than one certificate configured on the box (for example the self-generated trustpoint and the one pushed by DNA Center), it is strongly recommended you specify the certificate to be used for HTTPs access to the device. Not doing so may result in the Catalyst 9800 picking the wrong one and breaking access to the graphical user interface. In this case, use the following CLI command:

```
c9800(config)#ip http secure-trustpoint trustpoint-name
```

Alternatively, you can configure this in the GUI by going to the **Administration > Management > HTTP/HTTPS/Netconf** page and then selecting the specific certificate in the HTTP Trust Point Configuration section.

CI/CD Tools

Continuous integration and continuous development (CI/CD) are fundamental DevOps best practices for software development that emphasize automation to build and test software, focusing on achieving a software-defined lifecycle.

The Catalyst 9800 wireless controller is built on IOS-XE software, which provides open, standard-based programmable interfaces that support the integration of various CI/CD tools available in the market or allow you, as a network admin, to build your own.

Chapter 12, "Network Programmability," provides a technical description of the data models, protocols, and interfaces that allow C9800 to be programmatically managed.

Licensing

Cisco Smart Software Licensing makes it easier to buy, deploy, track, and renew Cisco software licenses for your Catalyst 9800. It allows you to create a pool of license resources that can be shared across multiple C9800 wireless controllers by removing older device-level entitlement and enforcement.

No licenses are required to boot up a C9800 wireless controller. However, each access point requires two licenses to be entitled to connect: one AIR Network License and one AIR DNA License. Both of these licenses can be configured to be either Essential or Advantage level. If there are not sufficient Cisco DNA licenses to cover all the access points connected to a Cisco Catalyst controller, an out-of-compliance message is displayed. This out-of-compliance message is purely informational and does not impact the functionality of the wireless deployment.

Starting with software release 17.3.2a, C9800 supports the Smart Licensing Using Policy, which is an enhanced version of Smart Licensing, with the overarching objective to further simplify the licensing solution:

- **Seamless Day 0 operations**: After a license is ordered, no preliminary steps, such as registration or generation of keys, are required unless you use an export-controlled or enforced license. There are no export-controlled or enforced licenses on Cisco Catalyst wireless controllers, and features can be configured on the device right away. There are no more evaluation licenses either.
- **Consistency in Cisco IOS-XE**: Campus and industrial Ethernet switching, routing, and wireless devices that run Cisco IOS-XE software have a uniform licensing experience.
- **Visibility and manageability**: Tools, telemetry, and product tagging are on the customer Smart Licensing Portal.
- **Flexible time-series reporting to remain compliant**: Easy reporting options are available, whether you are directly or indirectly connected to Cisco Smart Software Manager (CSSM), or in an air-gapped network.

For more information about licensing for the C9800, take a look at the frequently asked questions link found in the "References" section.

Cisco Next-Generation Wireless Stack

The Catalyst 9800 wireless controller is a key component of the next-generation wireless stack, as shown in Figure 1-17. Together with Cisco Catalyst Access Points, Cisco DNA Center, and DNA Spaces, it provides a complete end-to-end wireless solution for companies of all sizes, to manage the growing number of connected wireless devices and fulfill the evolving and growing digitization needs.

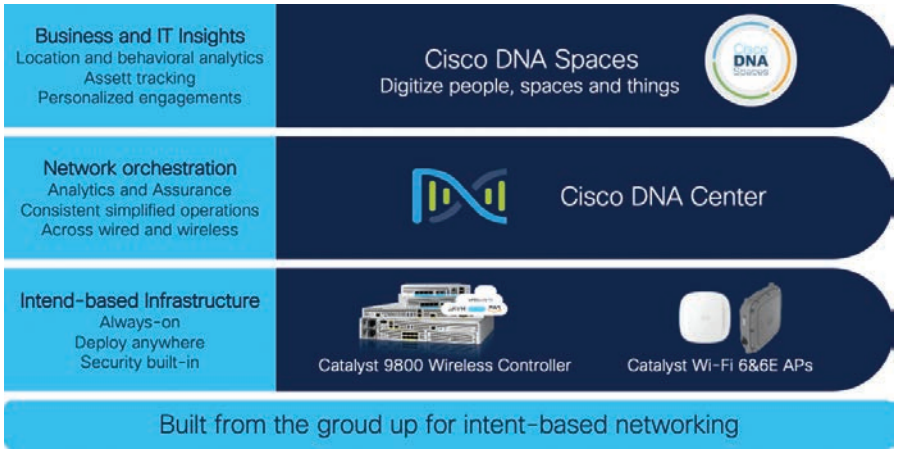


Figure 1-17 *Cisco next-generation wireless stack*

The Cisco next-generation wireless stack introduces wireless innovations at each and every layer, as illustrated in Figure 1-18.

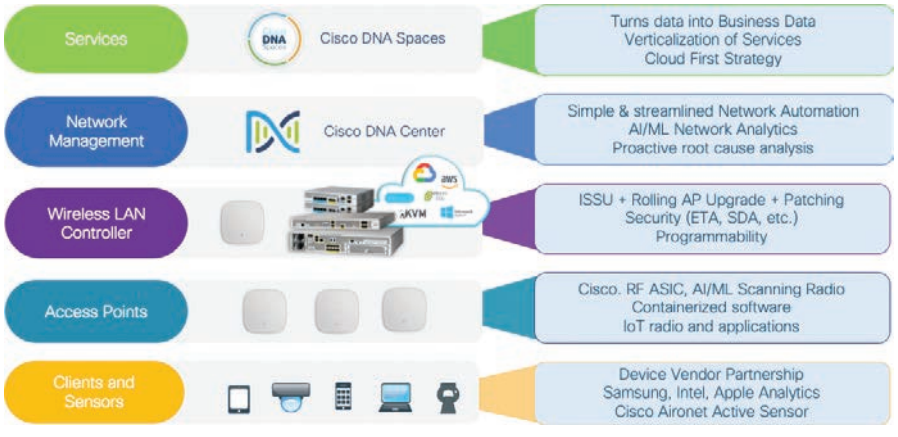


Figure 1-18 *Wireless innovation at each later of the stack*

Wireless innovations start at the device level with key partnerships with vendors like Apple, Samsung, and Intel; they allow the Cisco infrastructure to gain additional insights (iOS, Samsung, and Intel analytics) on device behavior, and provide additional quality of experience (for example, Apple Fastlane and Apple Fastlane+) so that these devices behave better on a Cisco wireless network. The Catalyst access points—thanks to Cisco RF ASIC, the IoT radio, and the IOx framework—go beyond the latest Wi-Fi 6 (802.11ax) standard and are ready for growing user expectations, IoT, and next-generation cloud-driven applications.

You previously learned about the advantages and innovations introduced by the Catalyst 9800 and Cisco DNA Center, so the last element is Cisco DNA Spaces. Whether it's learning more about visitors to your organization, your employees, or your things, such as assets and sensors, Cisco DNA Spaces digitizes your physical space. It does so by synthesizing location data across your sites to deliver location-based services at scale. This information can be used to enhance the customer experience, improve business operations and efficiencies (and reduce costs), realize industry-specific business outcomes, and much more.

Cisco DNA Spaces is a cloud-based platform for location: it leverages information from Wi-Fi to BLE tags, beacons, and other IoT sensors, and with gateway-enabled Cisco Wi-Fi 6 access points, it can easily scale advanced use cases while lowering total cost of ownership (TCO).

Summary

The Cisco Catalyst 9800 Series is the next generation of wireless LAN controllers from Cisco. It combines radio frequency (RF) excellence, gained over 20 years of leading the wireless industry, with Cisco IOS-XE software, a modern, modular, scalable, and secure operating system. This chapter introduces the Catalyst 9800, its benefits, and main characteristics, and clarifies the reasons for a new wireless controller for intent-based networking and the role it plays in a cloud-based management network. The chapter also describes the flexible options to manage the C9800, the licensing model, and how this new wireless controller fits into the next-generation wireless stack.

References

Feature Navigator tool: https://cfnng.cisco.com/mibs

Cisco Catalyst 9800 Series Configuration Best Practices: https://www.cisco.com/c/en/us/products/collateral/wireless/catalyst-9800-series-wireless-controllers/guide-c07-743627.html

Prime Infrastructure configuration guide: https://www.cisco.com/c/en/us/support/cloud-systems-management/prime-infrastructure/products-installation-and-configuration-guides-list.html

Licensing: https://cisco.com/go/smartlicensing

https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/17-3/config-guide/b_wl_17_3_cg/m-sl-using-policy.html

C9800 frequently asked questions: https://www.cisco.com/c/en/us/products/collateral/wireless/catalyst-9800-series-wireless-controllers/nb-06-cat9800-ser-wirel-faq-ctp-en.html

Cisco IOx framework in the Cisco IOS-XE and Linux OS integration: https://www.cisco.com/c/en/us/products/cloud-systems-management/iox/index.html

Chapter 2

Hardware and Software Architecture of the C9800

Hardware is often overlooked by software companies and vice versa. However, computer scientist Alan Kay said, “People who are really serious about making software should make their own hardware.” This is a principle dear to Cisco as well to other successful companies, such as Apple, that can leverage unique software capabilities thanks to their hardware. This chapter does not simply list numbers, cite specifications, and throw complex terminology at you but explains the important ideas that are the foundation of the Catalyst 9800 hardware and software architecture and its unique strengths. Let’s start with a quick introduction to the overall controller protocols architecture, which hasn’t changed much conceptually from the days of Cisco Unified Wireless.

General CAPWAP Split MAC Architecture

To this day, the access point (AP) software is still referenced on cisco.com as lightweight, and this makes a direct reference to the split from MAC architecture explained later in this section. There is no more “fat” (that’s really old naming) or “autonomous” software for wireless access points anymore (for 11ac Wave 2 APs and later), and they are always expected to join a wireless LAN controller (WLC) to operate. However, there are other modes of operation, such as Embedded Wireless Controller on AP (for Catalyst APs), where the AP acts both as an AP and a WLC (so it is autonomous in some way, but it still runs the lightweight software on the AP side of things), or the Workgroup Bridge mode, where the AP acts as a client to the infrastructure.

CAPWAP stands for Control and Provisioning of Wireless Access Points and is an IETF standard protocol. It builds two tunnels between an access point and the WLC: one for control and one for data. The control channel uses UDP 5246 port, and the data channel uses UDP 5247 on the WLC side (it’s the destination port when the AP sends traffic to the WLC, and the source port when the WLC sends traffic to the APs), in IPv4 or IPv6 (UDP Lite is used in IPv6 to save on processing of the checksum). APs use a random source port to send CAPWAP traffic to the WLC, and the WLC uses that same port to reply to APs.

The control protocol allows the WLC to centrally manage and configure all the access points and make sure they are always on the same software versions that correspond to the WLC version. It leverages Datagram Transport Layer Security (DTLS) for security and uses encryption of every packet.

The data protocol is an optionally encrypted way of tunneling back all the client data (including the initial onboarding and authentication, which are considered to be control-plane traffic from a controller handling perspective) to the controller to simplify the topology. Benefits of tunneling the traffic back to the controller are

- You don't need to span many VLANs to all the wired infrastructure hosting the APs. APs are connected on access ports in a management VLAN (typically dedicated to the APs).
- The WLC can host the client policies such as QoS or ACLs.
- The WLC is a central and unique point of contact for RADIUS and AAA authentication.
- Wireless clients' MAC and IP point of presence is behind the WLC data port from a wired network standpoint, which simplifies roaming by limiting MAC move events and lookups.

The split MAC architecture gets its name from the fact that the MAC layer responsibilities are split between the WLC and APs, as shown in Table 2-1.

Table 2-1 *Split MAC Architecture Responsibilities*

Access Point MAC Functions	Controller MAC Functions
802.11 beacons and probe responses (although probes are forwarded to the WLC as well)	802.11 associations requests, management and action frames
802.11 frame transmission and acknowledgments (including client power save handling and buffering)	802.11 QoS resource reservation
802.11 QoS frame queuing and packet prioritization	Client authentication in general
802.11 MAC layer data encryption and decryption	Client data traffic forwarding
Monitoring RF environment and scanning other channels	

Basically, all this boils down to a latency concern. It wouldn't be practical and efficient to send every frame to the controller to be decrypted, nor can the controller effectively manage the AP airtime and transmissions on the medium in real time. For everything else where there are benefits in centralizing the task on the controller and if latency is less of a concern, the task is best handled by the WLC.

This description also explains why there is a FlexConnect mode for APs in remote branches where the latency might be a bit higher and where you can optionally move some of the MAC functions back to the AP (like client authentication or data traffic forwarding).

DTLS is an IETF protocol based on TLS but targeted for use on UDP. All Cisco APs and appliance controllers are shipped with a manufacturer installed certificate (MIC), which is used to establish the CAPWAP DTLS tunnel and build the control tunnel encryption

key generation and mutual authentication. A locally significant certificate can also be generated and used for this purpose. CAPWAP control encryption is always used after the DTLS handshake occurs, but CAPWAP data packet encryption is an optional setting (automatically turned on in OfficeExtend mode). DTLS data encryption can have a performance impact on the global throughput numbers forwarded by the WLC, so enabling it when the data is transported over unsecured networks is advised. The data over the air is encrypted with the L2 security defined in the WLAN, if any (for example, WPA2-AES), and is always decrypted at the AP. At that stage, the AP encapsulates the wireless client frame in a UDP CAPWAP packet to send to the WLC. This client data could be perfectly readable and decodable in Wireshark if someone has access to the AP management network and collects a sniffer capture there. With CAPWAP data packet DTLS encryption, this CAPWAP payload would be encrypted, and the only thing someone capturing traffic would see is an encrypted CAPWAP data packet sent from a specific AP to the WLC without any clue with regards to what it contains.

The Controller Control Plane Architecture Elasticity

In the early codenames of the Catalyst 9800 during its inception, the word *elastic* was very present. Understanding why is the first step to understanding the whole architecture and its benefits.

IOS-XE Software Architecture

IOS-XE is based on a UNIX system named binOS internally (or Polaris as of its 16 and 17.x versions) and is a Cisco-modified version of UNIX. IOS (the now legacy one) was a monolithic operating system, a single process with a single memory space and fault domain. IOS-XE moves away from this architecture by adopting a multiprocess, modular, and scalable approach, separating the operating system (binOS) from the network tasks that are now managed by a process called IOSd. IOSd still takes care of the routing and interface configuration, but more specific tasks (like wireless tasks) are separated into dedicated processes. The management and config replication (in case of high availability) is also separated from the IOSd process. Figure 2-1 depicts the various processes in the IOS-XE architecture. Each of them is covered later in this chapter.

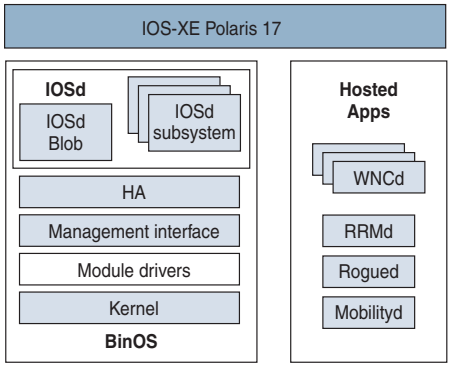


Figure 2-1 *IOS-XE general software architecture*

WNCd: The Heart of the Wireless Controller Control Plane

The Catalyst 9800 was designed with single-threaded event-driven processes in mind. The idea is to have as many processes as possible for the task while keeping those processes as independent as possible for session management purposes. Of course, processes can communicate with each other and they do, but vital activities around a specific given task are best handled by a unique process to avoid synchronization locks and wait times. There is no dynamic spawning of processes; they are statically spawned depending on the appliance hardware specifications. Similarly, multiple databases duplicate information for contention-free access: each process has its own database to which it has exclusive, contention-free access. These databases are duplicated into a central, consolidated database used for manageability and high availability replication. Thus, manageability does not impact session management and vice versa.

The databases are based on a versioned data model. This is a huge step forward compared to previous architectures, and this change (among a few others) is at the source of the programmability and the patchability of the problem. In Service Software Upgrade (ISSU), the new hitless upgrade system where you can upgrade your WLC high availability pair without any noticeable downtime in the network, is allowed by the fact that the WLC is based on modeled and versioned data. Having separate processes with their own databases and only minimal synchronization between processes and databases compared to the previous architecture, where all the threads were accessing the same central databases, avoids mutual exclusions and locks that prevent further scalability.

As depicted in Figure 2-2, the previous architecture relied on a single process running many threads fighting for a single memory space in a single fault domain for centralized data, causing contention. The Catalyst Wireless architecture, on the other hand, relies on several processes that are single-threaded and nonblocking. The number of Wireless Network Control daemon (WNCd) processes can be easily increased for scale, and there is not a single fault domain anymore (for example, memory separation). Although the increase in databases may seem like extra complexity, it is the conclusion of the platform being data model driven and able to easily externalize any data very quickly, boosting programmability in an efficient manner. The whole platform was also built with process patchability and restartability in mind.

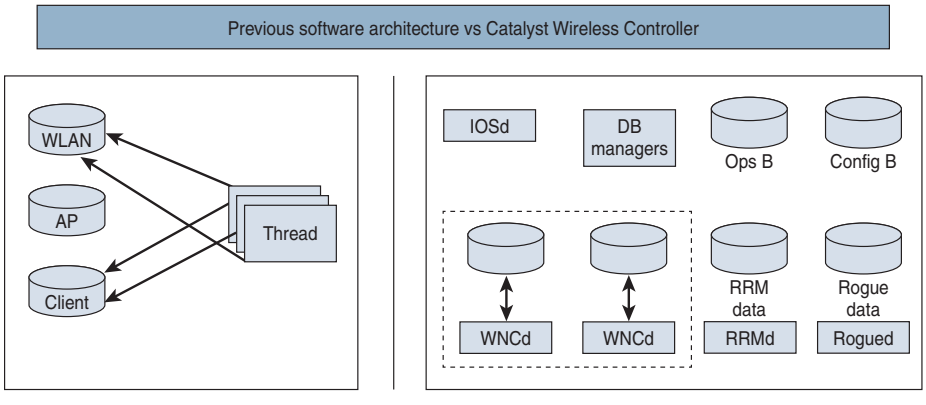


Figure 2-2 *The old wireless architecture on the left compared to the Catalyst Wireless architecture on the right*

The key wireless process is called Wireless Network Control daemon. The number of WNCd processes varies depending on the hardware that 9800 is running on (details are given in the "Hardware Overview" section). The WNCd process is a critical process managing APs and client sessions. Each WNCd process handles a specific set of access points and all the clients present on those access points; this maintains the approach where each process has everything it needs to handle a specific client session and keeps interprocess communication and replication to a minimum. The WNCd process is a single point for receiving and sending packets to the APs it manages but also implements a few other AP-facing capabilities like RRM client or probe handling.

This approach gives the vision of scalability of future Catalyst wireless platforms that will support an ever-increasing number of APs and clients simply by running more WNCd processes on more CPU cores. To oversee these processes, the WNCMgrd process manages the load-balancing of APs to WNCd instances. This means that the WNCMgrd is the one handling CAPWAP discoveries for the whole controller and assigns each new AP to a specific WNCd process (according to rules detailed later). The WNCMgrd is also in charge of centralizing information from each WNCd process in order to have a single go-to place for the "show wireless" CLI commands providing all the APs and clients information regardless of the number of WNCd processes. It has read-only access to the Centralized Wireless operational Database (CWDB), which contains all the real-time operational data. It can then consolidate information from each WNCd process and perform AP load-balancing and CLI information centralization tasks.

WNCd is a large process that is the center pillar of the Catalyst Wireless architecture (specific to the C9800) and contains many libraries inside it. You may hear a lot about SANET and SISF and see references to them in the logs and believe they are processes, but they are libraries inside the WNCd process. As a matter of fact, the general IOS-XE SANET library (in charge of AAA) has been copied (although modified) inside the WNCd process to manage the AAA authentication of wireless clients and their session management within the same process. Figure 2-3 depicts the various responsibilities of the SANET library that can be found inside a WNCd process.

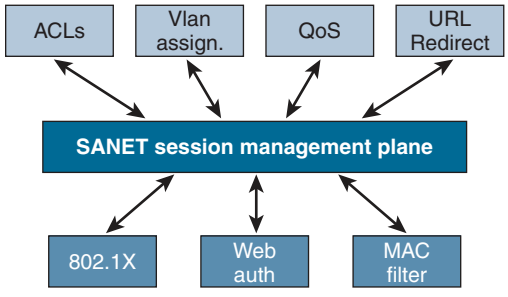


Figure 2-3 *SANET library responsibilities*

On top of that modified SANET library within WNCd, the Catalyst 9800 still has a SANET library inside the Session Manager Daemon (SMD) process just like other IOS-XE devices. That one handles wired session management (not really used in the Catalyst

9800) but also central Change of Authorization processing. Figure 2-4 shows the "classical" SANET library related to the SMD process in relation to the new SANET libraries inside the wireless processes.

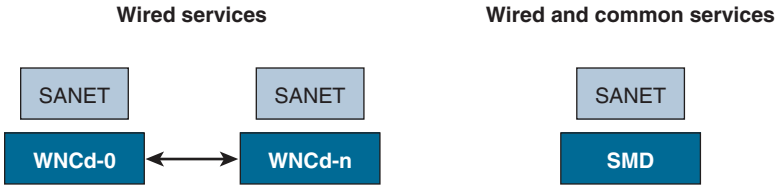


Figure 2-4 *WNCd processes and the AAA task split between the existing SMD process and the new SANET libraries inside WNCds*

Similarly, the Switch Integrated Security Features (SISF) library is integrated in the WNCd process to handle the wireless client DHCP or IP tracking process, among other things. SISF originated on the Catalyst switches to provide an integrated framework to process IPv6 control traffic as well as act on IPv6 data traffic. SISF responsibilities on the Catalyst 9800 include

- IPv6 NDP inspection (barring bogus NDP messages)
- NDP address gleaning: populating the binding table with information snooped in NDP traffic
- Device tracking
- IPv4 address gleaning: ARP and DHCP messages snooping
- DHCP relay with configured helper address
- NDP and ARP multicast suppression: unicasting NDP or ARP messages or responding on behalf of targets to save on broadcast/multicast traffic
- DAD proxy: duplicate address detection
- DHCP requirement: making sure the IP can only be learned through DHCP process

If you are accustomed to legacy Cisco Unified Wireless controllers, you might expect the Catalyst 9800 to do ARP proxy by default, but that is not the case. ARP proxy is a feature configurable in the policy profiles since IOS-XE 17.3 but is not enabled out of the box. The default behavior is for the 9800 to transform broadcast ARP messages destined to the wireless clients into a unicast message for the specific MAC address. This saves on airtime because the message does not have to be sent on all access points at the same time. It also brings efficiency because the destination client of the ARP request also learns about the source MAC at the same time. In other words, it saves one ARP request-reply exchange in the case of two wireless clients having to contact each other. However, it may cause spam or battery drain if a device on the network is constantly ARPing for one or more destinations. Enabling ARP proxy then allows the WLC to reply to the ARP request on behalf of the destination client without having bothered that client at all in the process (as long as it's a known and registered wireless client with the WLC).

A mobility agent library also takes care of the client roaming from inside the WNCd process. This strategy greatly reduces interaction with any other centralized process like IOSd over the interprocess communication (IPC) channel. Mobility is ensured no matter where the clients are hosted. A client being handled by an AP on one WNCd process can roam to an AP managed by another WNCd process without any noticeable delay; this is mostly a matter of internal process and resource optimization.

Other Wireless Processes

While WNCd is the biggest wireless process by far (because it contains many libraries as SANET and SISF, which were covered previously), there are smaller and more specific wireless-oriented processes, each with a role:

- Mobility daemon (Mobilityd) is a 9800-specific process handling all communications with mobility peers. It also maintains a PMK cache for roaming clients to have a centralized repository for those within IOS-XE. It knows a client's presence within the WLC to answer mobility peers accordingly during mobility announce messages.
- Radio Resource Manager daemon (RRMgrd) is a process that handles communications with Radio Resource Management (RRM) group members. It runs the RF grouping, TPC, DCA, and FRA algorithms. It talks to each RRM-client process in each WNCd.
- Rogued handles rogues, Wireless Intrusion Prevention System (WIPS), and Rogue Location Discovery Protocol (RLDP) functionalities. It gets rogue reports from WNCd processes and classifies the rogue APs and clients.
- NMSPd is a process handling all communications toward CMX or the DNA Spaces connector.
- Wstatsd handles Netflow (FNF) packets and maintains a database of AVC statistics.

Contrary to WNCd, all the preceding processes are single instances.

DBM is the process that handles the configuration database (which is used for WebUI querying), while ODM is the process that manages the CWDB (operational database). PubD is in charge of centralizing the information for model-driven access. Those are not C9800-specific processes but have important tasks in a daily C9800 operation.

Wireless Client State Machine

This discussion is all theoretical so far. A good way to see this information in action is to follow a client onboarding. Wireless clients associated with an AP are serviced by the WNCd instance that is also servicing that specific access point. This means there is a minimal dependency on other processes to complete a client join process. The client authentication is done by the SANET library inside the WNCd process; the client IP learning and DHCP phase are handled by the SISF library of the same WNCd process, and a mobility agent within WNCd handles any roaming event. In case the client

moves to an AP that is handled by another WNCd process (which should ideally not really happen; more on this in Chapter 3, "C9800 Configuration Model"), the mobility agent hands over the client to the other WNCd instance. There is no particular restriction to a client roaming between different WNCd processes. Client-related data is stored in separate database tables inside the respective WNCd process. A master table is available in the WNCmgrd to provide global tracking of clients wherever they are hosted or moved to. Each client is tracked by a finite state machine. The onboarding of a client is considered to be a Control Plane activity—that is, a task where traffic is analyzed and tracked by the Control Plane, and when the client moves to the final forwarding state of that state machine (called the RUN state), the traffic is forwarded by the datapath without mobilizing the Control Plane anymore. The client state machine's information is replicated between WLCs in a high availability pair of WLCs to guarantee a smooth failover.

The client state machine starts when an 802.11 management frame is received from the client (association request, reassociation request, 802.11 authentication frame). The client goes through various states until the RUN state where it is fully plumbed to the datapath. A client is deleted from the state machine based on certain timers (idle timeout, session timeout) or state timeout (after a certain time stuck in a transition state, the client is deleted). For example, if a client associates with a web authentication SSID (which is typically open) and does not log in on the web portal, the client entry is deleted after 5 minutes by default. This does not prevent the client from associating and trying again, but this really limits the overload and spam from clients who passively try to connect but do not finish the work and use resources (including IP address) for nothing.

Here is an example of an 802.1X client join state machine processing in a centrally switched WLAN configuration:

- The wireless client sends an association request, which is encapsulated in CAPWAP by the AP over the data tunnel to the controller.
- The WNCd (managing that AP) event library receives a socket data event.
- The CAPWAP data library inside the WNCd parses the CAPWAP data payload, classifies the payload as an 802.11 management payload, and invokes the registered handler function.
- The client orchestrator looks up an existing client state machine for this specific client based on the MAC address as unique identifier (which means that clients using random MAC and failing to associate and rotating their MAC address after each association attempt will create a lot of temporary client entries). If no existing state machine is found, it instantiates one and passes on the event to the 802.11 library to process the association request.
- The 802.11 library processes the association request and reads the BSSID operational table to validate the 802.11 parameters. It then sends an association response back. It invokes an internal API to send the association response over to the CAPWAP data tunnel.

- The client orchestrator sends an "add mobile" payload to the AP over the CAPWAP control tunnel to create the client entry on the AP itself. This is ACKed by the AP.
- The client orchestrator verifies the client L2 authentication policy configured. Because it is 802.1X authentication, it triggers the authentication interface state machine to start the Layer 2 authentication. A call to the SANET library is made.
- SANET takes care of the 802.1X exchange and forwards accordingly the packets encapsulated inside RADIUS to the RADIUS server up until an authentication failure or success.
- The SANET process writes the client policies, which may have been overridden by RADIUS if AAA override is configured. Those client policies are kept within SANET and available through internal API to other libraries.
- An IPC call is made to Mobilityd to update the PMK Cache key. It is maintained in Mobilityd at all times, even if there is no roaming to other WLCs. Mobilityd publishes the PMK key to the mobility group if any.
- The authentication interface triggers the four-way key exchange to generate the PTK, GTK, and IGTK. The authentication interface gets back to the client orchestrator with the result of the Layer 2 authentication.
- The client orchestrator then sends the transient keys to the AP in another "Add mobile" payload inside CAPWAP control.
- If there are controllers in the mobility group, the Mobility interface starts the processing by announcing the client and finds out if the client already exists on another WLC (and is therefore roaming between different WLCs). The WLCs then negotiate the exchange of the client information.
- The client orchestrator transitions to the "IP learn" state and triggers the IP learn library (called SISF) inside WNCd to learn the IP address of the client.
- The IP learn interface state machine plumbs a client entry into the datapath. This allows the datapath to learn about the client MAC and punt all the DHCP payloads in the client VLAN to the WNCd process (that is, the DHCP packets are taken out of the dataplane and sent to the CPU for further processing and learning).
- When the client receives its IP from the DHCP server, the SISF library identifies the IP address, creates an event, and returns the information to the client orchestrator.
- The client orchestrator drives the client policy bind to the datapath by providing all the policies such as QoS, ACL, VLAN, and IP.
- The client orchestrator sends the final "add mobile" payload with the QoS attributes to the AP and then transitions the client to RUN state.

- From this point onward, the client traffic is forwarded according to policies and will be until a timeout (session or idle for example) is hit or until another client management event is triggered (roaming, new association, and so on).

The controller needs to learn about at least one IP address used by the client but supports up to eight addresses learned for the same MAC (useful in IPv6). The preceding is a detailed explanation of the state machine changes in plain English. It may help to see a diagram showing the state transitions. Figures 2-5 and 2-6 depict the same overall client state machine upon connecting a new wireless client to a WPA2 Enterprise SSID until the client is placed in the RUN state. The client states are written as they are found in RadioActive tracing logs (for more details, see Chapter 16, "Troubleshooting").

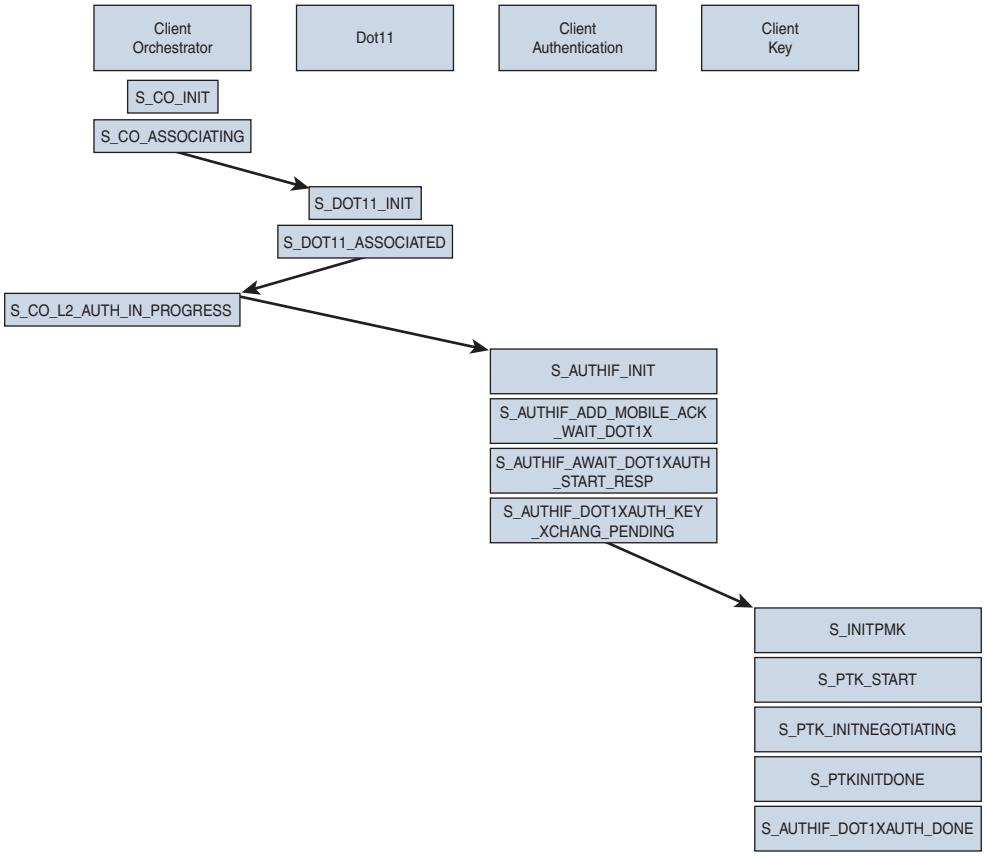


Figure 2-5 *Client state machine on a WPA2 Enterprise SSID*

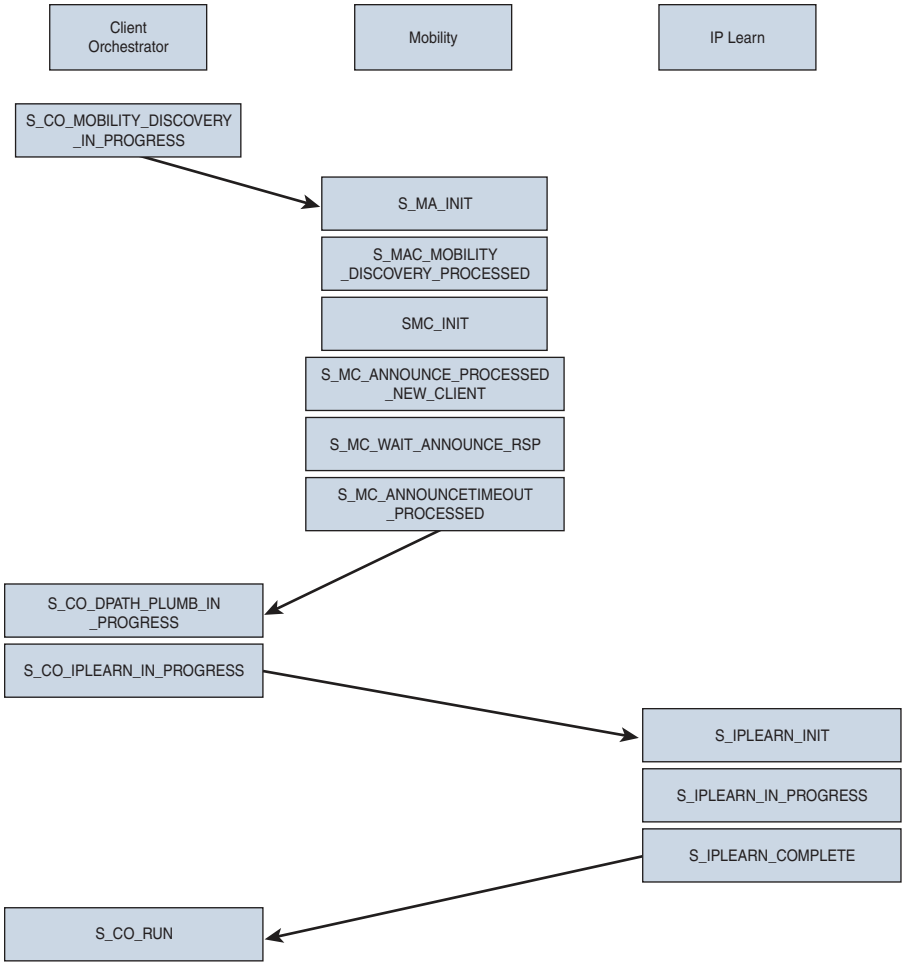


Figure 2-6 *Client state machine on a WPA2 Enterprise SSID (continued)*

One Dataplane to Rule Them All (or Three at the Maximum)

One of the great achievements with IOS-XE 17 (although it started in the IOS-XE 16 train) is to unify the software for routers, switches, and now wireless controllers too. The goal is to run the same IOS-XE on all the enterprise network devices. It would be easy to say that the Catalyst 9800 is very similar to the rest of the Catalyst switches, but that is not the case although they share the IOS-XE operating system and a somewhat similar process structure. After all, access points are also part of the Catalyst family now, and they don't run the same operating system at all. The dataplane is one big difference, and this is why you cannot use the same software file to install on a Catalyst 9300 and on