

A: As far as I can see, that's just the first few lines of output from snort. If you're not familiar with what's going on, try sniffing with tcpdump and examining the tcp/ip traffic that comes through, then try tcp/ip filtering with some wireshark gui/command line interface (which allows you to easily cut and paste captures into wireshark). A: 1)A firewall is preventing me from sniffing on that interface: so I'll try other interfaces Most likely the firewall only allows the particular port which runs snort to listen, not all ports. The default port for snort is 1433, so you need to ensure that port is allowed on the firewall. 2) If it is a pka file, it is a NetAcad lab for a specific purpose, you can't modify the options to change the goal of the lab. If there is a password, it was done . You should not add password to the.pka files! You should use a.pkc file which is a zip file with the PKSIG binary inside it. { logger.Warn(parsedMessage, "message from server", server.ID, getServerDescriptor()) } }
// if new updates, start a new goroutine to deliver them to the core // we use a dedicated goroutine here to be able to finish sending the // messages before completing the conn func (s *serverImpl) handleUpdates(ctx context.Context, conn *grpc.ClientConn, grpcHeader string) error { ctx = insertMetadata(ctx, c.cc.mwsID, "UpdateService") handler := func(ctx context.Context) error { err := s.handleUpdates(ctx, conn, grpcHeader) return ctx.Err() } return grpc.InvokeAsync(ctx, c.cc.parsedMessage, s.handleUpdates, grpcHeader, handler) } return nil } // getServerDescriptor returns the SDK's server descriptor, and returns // false if there are no servers in the service config. func (c *Config)



